Univerza v Ljubljani
Fakulteta za računalništvo in informatiko

Tadej Magajna

Izbira značilk pri večslojnem gručenju

magistrsko delo

študijski program druge stopnje
računalništvo in informatika

Mentor: izr. prof. dr. Marko Robnik-Šikonja

Ljubljana, 2016

Zahvala

Posebna zahvala gre mentorju izr. prof. dr. Marku Robniku-Šikonji za pomoč, potrpežljivost in strokovno vodenje. Kot soavtor ene glavnih tehnik izbire značilk je nudil koristne nasvete, ki so močno pripomogli k uspešnosti naloge. Zahvaljujem se tudi prof. dr. Mari Bresjanac Blinc za strokovno mnenje s področja nevrologije.

Tadej Magajna, 2016

Kazalo

Seznam uporabljenih kratic

kratica	angleško	slovensko
mvReliefF	Multi View ReliefF	ReliefF z večimi pogledi
mvReliefF-md	Multi View ReliefF - Multi Distance	ReliefF z večimi pogledi - z večimi razdaljami
mvReliefF-mh	Multi View ReliefF - Multi Hit	ReliefF z večimi pogledi - z večimi zadetki
mvReliefF-mdmh	Multi View ReliefF - Multi Distance, Multi Hit	ReliefF z večimi pogledi - z več razdaljami in več zadetki
ADNI	Alzheimer's Disease Neuroimaging Initiative	Iniciativa za Alzheimerjevo bolezen
ARI	Adjusted Rand Index	popravljen Randov indeks
NMI	Normalized mutual information	normalizirana medsebojna informacija
RIPPER	classification rule learning algorithm (Repeated Incremental Pruning to Produce Error Reduction)	algoritem za učenje odločitvenih klasifikacijskih pravil (inkrementalno rezanje z zmanjševanjem napake)
CSPA	Cluster-based similarity partitioning algorithm	ločevanje na podlagi podobnosti za gručenje

Povzetek

Naslov: Izbira značilk pri večslojnem gručenju

V nalogi predstavimo področje izbire značilk pri večslojnem gručenju. Opišemo učenje z večimi pogledi in večopisno gručenje. Predlagamo novo metodologijo gručenja z večimi pogledi z opisom gruč, katerega rezultat so gruče, opisane na več načinov. Tovrstni opisi predstavljajo človeku razumljive razlage gruč in nudijo lažje razumevanje povezav med pogledi. Na umetni množici pokažemo, da predlagana tehnika izbire značilk mvReliefF uspešno deluje na podatkovnih zbirkah za učenje z več pogledi. Na podatkovni zbirki iz UCI repozitorija primerjamo dobljene rezultate s sorodnim člankom, kjer naši rezultati kažejo na izboljšanje uspešnosti gručenja. Metodologijo izvedemo na podatkih ADNI pacientov z Alzheimerjevo boleznijo. Dobljene gruče s tehnikami razlage prediktorjev opišemo posebej s kliničnimi in posebej z biološkimi značilkami. Te predstavljajo človeku razumljive razlage gruč in povezav med kliničnimi in biološkimi značilkami. Nevrološka analiza potrjuje smiselnost dobljenih gruč in povezav med sicer znanimi značilkami obeh pogledov.

Ključne besede: učenje z večimi pogledi, večopisno rudarjenje, izbira značilk, algoritem ReliefF, Alzheimerjeva bolezen.

Abstract

Title: Feature Selection for Multilayer Clustering

We present an overview of feature selection for multi-layer clustering. We explain the concepts of multi-view learning and redescription mining. We propose a new clustering method using predictor explanations which provide multiple explanations for each resulting cluster. These explanations serve as a interpretable definition of groups and can help to understand connections between features from different views. Test on our synthetic data set shows that the proposed multi-view feature selection method mvReliefF handles multi-view data well. On a data set from UCI repository we compared our method with published results. On a joined ADNI Alzheimer's disease data set, we explain the obtained clusters separately with clinical and separately with biological features using predictor explanations. The explanations serve as an interpretable cluster definitions and help to understand the connections between clinical and biological features. Neurological analysis suggests that the obtained clusters and connections between view features are meaningful.

Keywords: multi-view learning, redescription mining, feature selection, ReliefF algorithm, Alzheimer's disease.

Poglavje 1

Uvod

Gručenje ali rojenje je tehnika razvrščanja elementov v skupine. Cilj gručenja je izbrati take skupine, da so si elementi znotraj skupin čim bolj podobni in elementi med skupinami čim bolj različni. V dveh dimenzijah si lahko postopek gručenja predstavljamo kot iskanje krogov, ki obkrožajo skupine elementov. Z dodajanjem novih dimenzij se količina informacij v podatkovni zbirki v principu veča, a se prav tako veča tudi zahtevnost učenja in verjetnost, da nekatere značilke slabo ločujejo gruče. Takšne značilke lahko poslabšajo uspešnost gručenja, zato v strojnem učenju uporabljamo metode za izbiro značilk, s pomočjo katerih lahko odstranimo nepomembne značilke. Večslojno gručenje (angl. multiview clustering) je tehnika strojnega učenja, ki izkorišča dejstvo, da so problemi pogosto opisani na več načinov, z večih pogledov. Obstoječe metode večslojnega gručenja prinašajo dobre rezultate, a standardne tehnike strojnega učenja, kot je izbira značilk, še niso prilagojene za tovrstno učenje. Večslojno gručenje se uspešno uporablja na medicinskih podatkih za diagnozo in preprečevanje bolezni [4, 48], saj so ti pogosto opisani z večimi pogledi in so preobsežni za ročno označevanje. Ena kritičnih, trenutno še neozdravljivih bolezni, je Alzheimerjeva bolezen. Statistična raznolikost, število neinformativnih značilk in šum v podatkih Alzheimerjeve bolezni predstavljajo izziv za učenje z večimi pogledi in za uporabo novih metod za izbiro značilk pri večslojnem gručenju.

1.1 Učenje z večimi pogledi

Podatki v strojnem učenju so pogosto opisani z značilkami z različnih virov ali pa so pridobljeni z različnimi metodami za izločanje značilk. Te skupine značilk, ki si pogosto delijo podobne statistične lastnosti, imenujemo pogledi. Pri konvencionalnih metodah strojnega učenja značilke iz večih pogledov tipično združimo v en pogled, ali pa uporabimo le značilke iz posameznega pogleda in se s tem izgubino določen del informacije [48]. Večslojno gručenje je metoda strojnega učenja z večimi pogledi, ki problem ocenjevanja gručenj rešuje s predpostavko, da je kvaliteta gručenja odvisna od stopnje ujemanja gručenj z večih pogledov. Kljub temu, da je metod za klasifikacijo in gručenje z večimi pogledi veliko in prinašajo dobre rezultate, je proces izbire značilk pri učenju z večimi pogledi manj raziskano področje. Prav tako večina algoritmov za izbiro značilk temelji na učenju z enim pogledom. Tako se pojavi potreba po razvoju novih oziroma adaptaciji obstoječih algoritmov za izbiro značilk za učenje z večimi pogledi.

1.2 Alzheimerjeva bolezen in ADNI

Alzheimerjeva bolezen je nevrodegenerativna bolezen možganov, ki predstavlja najpogostejši tip demence. Vpliva na sposobnost govora, spomin, orientacijo in splošne kognitivne sposobnosti pacienta. Ločimo tri oblike bolezni: blaga, zmerna in težka. Bolezen še ni ozdravljiva, saj nobena od ustaljenih tehnik zdravljenja ne ustavi popolnoma bolezenskega procesa. S podaljševanjem življenjske dobe postaja Alzheimerjeva bolezen ena najbolj kritičnih bolezni staranja. Z namenom razvoja diagnoze, preprečevanja nastanka in zdravljenja bolezni je bil začet projekt Alzheimer's Disease Neuroimaging Initiative (ADNI), ki hrani podatke številnih pacientov z Alzheimerjevo boleznijo v različnih fazah bolezni. Ponuja obširno podatkovno zbirko lastnosti pacientov, ki so opisane s kliničnimi in biološkimi značilkami. Tovrstna razdelitev podatkov predstavlja primerno podatkovno zbirko za učenje z večimi pogledi, odvečni in šumni podatki pa potrebo po izbiri značilk [30].

Naloga predstavi področje učenja z večimi pogledi s poudarkom na večslojnem gručenju. Predstavimo novo metodo za izbiro značilk pri večslojnemu gručenju, ki temelji na prilagoditvi algoritma ReliefF [38]. Uspešnost preizkusimo na umetnih podatkih in na javnih podatkovni zbirki ter primerjamo uspešnost z rezultati iz sorodnih člankov. Nazadnje metodo izvedemo na ADNI podatkih. Rezultate ovrednoti strokovnjakinja s področja nevrologije.

1.3 Napoved vsebine

Delo je sestavljeno iz šest poglavij in dodatkov. V poglavju 2 je podrobneje razloženo učenje z večimi pogledi in problematika gručenja. V 3. poglavju so predstavljene glavne podatkovne zbirke in problematika Alzheimerjeve bolezni. V 4. poglavju je predstavljena predlagana metodologija izbire značilk pri večslojnem gručenju. V 5. poglavju opišemo metodologijo in rezultate empirične evalvacije. Delo se zaključi s 6. poglavjem, kjer so navedene sklepne ugotovitve ter ideje za izboljšave in nadaljnje delo. V dodatkih so opisi značilk ADNI podatkovne zbirke, vizualizacije razlag gruč bolnikov z Alzheimerjevo boleznijo in opis doprinosa k odprtokovnim knjižnicam za izbiro značilk.

Poglavje 2

Pregled področja

Izbira značilk pri večslojnem gručenju je tehnika strojnega učenja, ki združuje učenje z večimi pogledi, gručenje, izbiro značilk in kriterije za oceno gručenja. Za celovito razumevanje problema v poglavju sprva predstavimo principe učenja z večimi pogledi in pomembnejše pristope. Opišemo osnovne metode gručenja in jih primerjamo z ostalimi tehnikami strojnega učenja. Predstavimo glavne metode za izbiro značilk in razložimo njihove lastnosti. Podrobno razložen algoritem ReliefF. Nazadnje predstavimo področje razlage prediktorjev, ki je ključnega pomena pri kompleksnih problemih, kjer je potrebna človeška interpretacija (npr. medicinska diagnoza).

2.1 Učenje z večimi pogledi

Na mnogih področjih uporabe strojnega učenja značilke zajemamo z različnih domen in z različnimi metodami za izločanje značilk. Učenje z večimi pogledi iz vsakega pogleda izlušči znanje, ki mu je specifično, zmanjša pojav prekomernega prileganja in izboljša klasifikacijsko točnost. Blum in Mitchell (1998), začetnika učenja z večimi pogledi, kot primer učenja z večimi pogledi v članku [3] navedeta učenje o vsebini spletnih strani, kjer besedilo spletnih strani predstavlja en pogled in besedilo hipertekstnih povezav na straneh pa drug pogled. Trenutne algoritme, ki delujejo na principu učenja z večimi po-

gledi, lahko razdelimo v tri kategorije: soočenje (co-training), učenje z večimi jedri in učenje na podlagi podprostorov [48].

Metode učenja z večimi pogledi delimo na tri kategorije glede na fazo kombiniranja pogledov. *Zgodnja kombinacija* pogledov je prisotna, ko se značilke iz različnih pogledov že v začetni fazi učenja združijo v eno podatkovno zbirko. Ta pristop uporablja učenje na podlagi podprostorov. *Vmesna kombinacija* se izvaja pri učenju z večimi jedri, saj se kombinacija jedrnih funkcij izvaja v vmesni fazi učenja. *Pozna kombinacija* pogledov se izvaja pri pristopih, kjer se klasifikatorji ali gručenje izvaja na značilkah vsakega pogleda posebej in se ujemanje rezultatov med pogledi vrednoti šele v zadnji fazi učenja. To je značilno za algoritme soočenja [48, 32].

2.1.1 Soočenje

Soočenje je področje strojnega učenja z večimi pogledi, ki temelji na principu konsenza [8] med hipotezami, zgrajenimi na podatkih iz različnih pogledov. Učna množica je sestavljena iz enega ali večih pogledov. Učno množico sestavlja manjša množica označenih podatkov, ki so tipično manj dostopni in pogosto označeni s človeško pomočjo, in obsežnejše množice neoznačenih podatkov, ki so cenejši in dostopnejši. Tako je soočenje vrsta delno nadzorovanega učenja [50]. Učno množico X lahko predstavimo kot množico značilk $X = X_1 \frown X_2$, kjer X_1 in X_2 predstavljata prvi in drugi pogled. Algoritmi soočenja sprva na podlagi manjše množice označenih podatkov naučijo dva modela h_1 in h_2, kjer se h_1 uči na označenih podatkih X_1 in h_2 na označenih podatkih X_2. Zaradi omejenega števila označenih podatkov je klasifikacijska točnost tipično nizka, nestrinjanje med hipotezami pa visoko. Nato podmnožico neoznačenih podatkov X označimo s pomočjo modela h_1 in jih dodamo v učno množico h_2. Prav tako s pomočjo h_2 označimo podmnožico neoznačenih podatkov, ki jo dodamo učni množici h_1. Modela se tako izmenoma učita na novo označenih podatkih in izmenjujeta informacije med pogledi, dokler konsenz med modeloma in klasifikacijska točnost na te-

stni množici ne doseže optimuma [3, 48].

Posebna vrsta algoritmov soočenja so soregularizacijski algoritmi, ki z uporabo regularizacije z večimi pogledi za oba pogleda razvijejo modela h_1 in h_2, ki v splošnem rešujeta naslednji regularizacijski problem:

$$\min \sum_{i \in N} [h_1(x_i) - h_2(x_i)]^2 + \sum_{i \in O} V(y_i, f(x_i)), \tag{2.1}$$

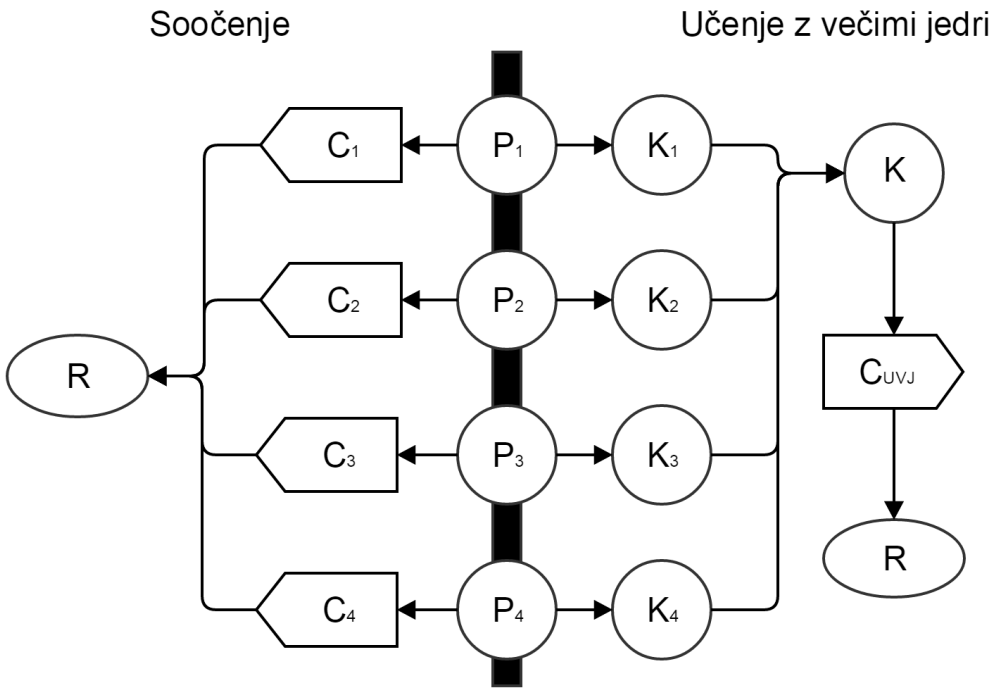
kjer V predstavlja funkcijo izgube (klasifikacijske napake) in y_i razred i-tega primera. Prvi seštevanec predstavlja stopnjo neujemanja rezultatov na označenih podatkih O in drugi stopnjo neujemanja na neoznačenih podatkih N [48]. V članku [43] je opisana razširitev konvencionalnih klasifikacijskih algoritmov z uporabo soregularizacijskih funkcij. Empirične raziskave pokažejo izboljšanje klasifikacijske točnosti pri razširitvi metode podpornih vektorjev.

Posebna tehnika algoritmov soočenja je tudi večslojno gručenje, ki jo natančneje razložimo v razdelku 2.5

2.1.2 Učenje z večimi jedri

Uporaba jedrnih funkcij pri algoritmih v strojnem učenju je pogosta [35]. Znan primer je metoda podpornih vektorjev (support vector machines - SVM) [21]. Ta je v osnovi linearen klasifikator, ki lahko z uporabo jedrnega trika rešuje nelinearne probleme. Pri jedrnih metodah z enim jedrom ustrezno funkcijo jedra in optimalne parametre jedra tipično izberemo s pomočjo prečnega preverjanja [23]. Metode učenja z večimi jedri (UVJ) pa z uporabo linearne ali nelinearne kombinacije večih funkcij jedra poiščejo optimalno kombinacijo funkcij jedra in njihovih parametrov. Zaradi lastnosti aditivnosti jeder je kombinacija jedrnih funkcij prav tako jedrna funkcija [29]. Uporaba večih jeder pri učenju z večimi pogledi pride do izraza pri učenju na podatkih, kjer imajo pogledi različne statistične lastnosti. Dober primer je podatkovna zbirka opisana s slikami in zvokom videa, saj zvok in slike uporabljajo različne

mere podobnosti in tako ustrezajo različnim jedrnim funkcijam. Tako je pri učenju z večimi pogledi smiselna uporaba kombinacije večih jedrnih funkcij [20].



Slika 2.1: Primerjava soočenja in učenja z večimi jedri. Vozlišča $P_{1,..4}$ predstavljajo različne interpretacije podatkov, poglede. Vozlišča $C_{1,..4}$ predstavljajo klasifikatorje, kjer je C_{UVJ} klasifikator primeren za učenje z večimi jedri. Vozlišča $K_{1,..4}$ predstavljajo jedra in K kombinacijo jeder. Vozlišče R predstavlja končni rezultat.

Linearne metode kombinacij jeder iščejo optimalno kombinacijo z uporabo uteži $\{w_k\}_{k=1}^{M}$. Posamezno jedrno funkcijo K_k lahko predstavimo kot $\mathbb{R}^D \times \mathbb{R}^D x \mapsto \mathbb{R}$ in $\{K_k\}_{k=1}^{M}$ kot vektor vseh jedrnih funkcij. Najpreprostejši pristop

linearnih kombinacij je jedro direktnega seštevanja:

$$K(x_i, x_j) = \sum_{k=1}^{M} K_k(x_i, x_j), \tag{2.2}$$

ki ima vrednosti vseh uteži postavljene na 1. Ker pa so vse jedrne funkcije redko prava izbira, je smiselna uporaba jedra uteženega seštevanja:

$$K(x_i, x_j) = \sum_{k=1}^{M} w_k K_k(x_i, x_j). \tag{2.3}$$

Poleg omenjenih pristopov se pri učenju z večimi jedri uporabljajo tudi nelinearne kombinacije jedernih funkcij, a empirične raziskave ne kažejo opaznega izboljšanja [48].

2.1.3 Učenje na podlagi podprostorov

Učenje na podlagi podprostorov je metoda nenadzorovanega učenja za zmanjšanje dimenzionalnosti podatkov. Izražanje visokodimenzionalnih podatkov v nižje dimenzijskem prostoru se uporablja za odkrivanje latentnih informacij [1], vizualizacijo podatkov in izboljšanje klasifikacijske točnosti in gručenja [47]. Znani metodi učenja na podlagi podprostorov sta metoda glavnih osi (Principal Component Analysis - PCA) in metoda linearne razločevalne analize (Linear discriminant analysis)[47].

Rezultat učenja z več pogledi na podlagi podprostorov je latenten podprostor, za katerega se predpostavlja, da so bili iz njega generirani originalni pogledi. Znana izpeljanka PCA za učenje z večimi pogledi je kanonična korelacijska analiza (canonical correlation analysis - CCA). Ta skuša najti tak set baznih vektorjev, da je linearna korelacija med projekcijama na te bazne vektorje maksimalna [46]. Slabost CAA je, da zna iskati samo linearne korelacije med pogledi, tako se za nelinearne probleme uporablja izpeljanka

[1] prikrite, ne direktno razumljive informacije

jedrna kanonična korelacijska analiza (kernel canonical correlation analysis - KCCA) [48].

2.1.4 Ustreznost za učenje z večimi pogledi

Razčlenitev značilk podatkovne množice na več pogledov je lahko naravna (značilke različnih pogledov so zajete z različnih domen) ali umetna (značilke se razdelijo na več pogledov naknadno). Ustreznost skupin značilk za uspešno učenje z večimi pogledi je odvisna od treh glavnih kriterijev [3, 48]:

1. *Zadostnost:* vsak pogled je sam zadosten za uspešno klasifikacijo
2. *Kompatibilnost:* modeli z obeh pogledov z veliko verjetnostjo vrnejo podobne rezultate
3. *Pogojna neodvisnost:* pogledi so pogojno neodvisni za določen razred

Prvi pogoj izvira iz predpostavke, da pogled, ki ne nosi informacije o razredu pri klasifikaciji, ne bo prispeval k učenju z večimi pogledi. Pogoj zahteva, da je stopnja napake P_{err} vsakega pogleda relativno majhna.

Drugi pogoj zahteva, da se rezultati obeh pogledov vsaj z določeno verjetnostjo ujemajo. Zaradi principa konsenza [8] je drugi pogoj tesno povezan s prvim. Dasgupta in sod. (2002) predstavijo enega temeljnih principov učenja z večimi pogledi, ki temelji na predpostavki, da je kvaliteta učenja z večimi pogledi odvisna od stopnje ujemanja rezultatov hipotez posameznih pogledov. Povezava med verjetnostjo neujemanja rezultatov hipotez in njihovo klasifikacijsko točnostjo je predstavljena z enačbo:

$$P(h_1 \neq h_2) \geq max\{P_{err}(h_1), P_{err}(h_2)\}. \tag{2.4}$$

Neenačba trdi, da verjetnost nestrinjanja hipotez $P(h_1 \neq h_2)$ predstavlja zgornjo mejo verjetnosti napake obeh hipotez. Enačba tvori povezavo med prvim in drugim pogojem za ustreznost učenja z večimi pogledi.

Tretji pogoj zahteva, da so pogledi med seboj pogojno neodvisni. Glede na članek [11] sta dogodka x in y pogojno neodvisna pri dogodku z, če sta

prisotnost ali odsotnost dogodka x in prisotnost ali odsotnost dogodka y neodvisna dogodka. Pogojna neodvisnost je formalno definirana z ekvivalentnimi enačbami :

$$\begin{aligned} p(x, y|z) &= p(x|z)p(y|z), \\ p(x|y, z) &= p(x|z), \\ p(y|x, z) &= p(y|z). \end{aligned} \tag{2.5}$$

V primeru učenja z večimi pogledi x in y predstavljajo značilke iz posameznih pogledov in z razred, ki pripada učnim primerom [3]. Izkaže se, da je ta za realne podatke prestrog, saj je večina pogledov iz podatkovnih zbirk realnih podatkov vsaj do neke mere pogojno odvisnih. Tako pogosto tretji pogoj v praksi nadomesti manj stroga predpostavka, da značilke med pogledi niso popolnoma korelirane [3].

2.2 Uvod v gručenje

Strojno učenje se v grobem deli na nadzorovano in nenadzorovano. Pri nadzorovanem učenju se algoritmi učijo na podlagi vhodnih podatkov in pripadajočih oznak razredov. Rezultat je napovedni model, katerega namen je napovedati oznake razredov za nove, neoznačene primere. Najpogostejša tehnika nadzorovanega strojnega učenja je klasifikacija. Ta se pogosto sooča s problemom prekomernega prileganja učnim podatkom, saj se napovedni model preveč prilagodi učnim podatkom, ki pogosto vsebujejo šum, in tako novih primerov ne klasificira ustrezno. Tako je pri klasifikaciji pomembno pravo razmerje med generalizacijo in prileganjem učnim podatkom z uporabo regularizacijskih metod. Klasifikacijska točnost se tipično ovrednoti na testni množici, ki je ločena od učne [17]. Ker se zaradi testne množice algoritem posledično uči na manjši množici podatkov, se uporablja prečno preverjanje. Ta v vsaki iteraciji razdeli podatkovno zbirko na testno in učno množico in preveri klasifikacijsko točnost modela. Povprečenje rezultatov vseh iteracij prinese natačnejšo oceno točnosti modela, ki je manj občutljiva

na prekomerno prileganje [23].

Nenadzorovano učenje, ki ga pogosto imenujemo tudi gručenje ali rojenje, temelji na učenju na neoznačenih podatkih. Sooča se s problemom ocenjevanja točnosti gručenja. Pri nadzorovanem učenju je ocena točnosti v osnovi trivialen problem, ki temelji na primerjavi dobljenih napovedi s pripadajočimi oznakami razredov. Ker pri nenadzorovanem učenju oznak razredov ni, se pa za oceno uporabljajo notranji in zunanji kriteriji, ki jih opišemo v razdelku 2.3. Kljub temu je nenadzorovano učenje priljubljena tehnika, saj so neoznačeni podatki cenejši in bolj dostopni, saj označevanje pogosto zahteva strokovno znanje in ročno delo.

Nadzorovano in nenadzorovano učenje se v kombinaciji uporabljata pri delno nadzorovanem učenju. Ta temelji na kombinaciji nenadzorovanega učenja na (bolj dostopnih) neoznačenih podatkih in nadzorovanega učenja na (tipično manjši, manj dostopni) množici označenih podatkov. Tipičen primer klasifikacijskih tehnik delno nadzorovanega učenja so klasifikatorji soočenja iz razdelka 2.1.1.

Najpogostejša tehnika nenadzorovanega učenja je gručenje. Namen gručenja je razvrščanje elementov v skupine, ki jih imenujemo gruče. Matematična razlaga gručenja je sledeča: $X \in R^{m \times n}$ je podatkovna zbirka m elementov x_i opisanih z n značilkami. Cilj gručenja je razdeliti X na K skupin C_k tako, da so si elementi, ki spadajo v iste skupine, bolj podobni kot elementi z različnih skupin. Rezultat algoritma je injektivna preslikava $X \to C$ elementov x_i v gruče C_k [17]. Število $K \in \mathbb{N}$ je glede na algoritem določeno vnaprej ali ga pa izbere algoritem sam [17].

2.2.1 Metode gručenja

Metod gručenja je veliko, saj se ne razlikujejo le po metodi iskanja optimalne uvrstite elementov, temveč tudi po razlagi ločenosti in prekrivanja gruč. Tako jih lahko glede na razdeljevanje delimo v naslednje kategorije:

- strogo ločeno razdeljevanje,

- strogo ločeno razdeljevanje z osamelci,
- hierarhično gručenje,
- gručenje s prekrivanjem.

Pri strogo ločenem razdeljevanju lahko posamezen element spada v samo eno gručo. Strogo ločeno razdeljevanje z osamelci je enako, le da določeni elementi ne pripadajo nobeni gruči. Imenujemo jih osamelci. Pri gručenju s prekrivanjem se tipično izvaja več gručenj, kjer se gruče med seboj delno ali v celoti prekrivajo. Tipičen primer gručenja s prekrivanjem je večslojno gručenje opisano v razdelku 2.5.

Glede na metodo povezovanja in razvrščanja točk poznamo v grobem naslednje kategorije gručenj:

- povezovalno gručenje,
- gručenje na podlagi centroidov,
- gručenje na podlagi gostote,
- verjetnostno gručenje.

Povezovalno gručenje temelji na razdaljah med posameznimi pari elementov [42]. Primer povezovalnega gručenja je hierarhično gručenje. Rezultat je drevo, ki se imenuje dendrogram. Vozlišča v dendrogramu predstavljajo gruče. Vozlišča imajo lastnost, da vsi elementi, ki spadajo v gruče otrok vozlišča, spadajo tudi v gručo starševskega vozlišča [2].

Pri gručenju na podlagi centroidov, so gruče definirane s centralnim vektorjem, ki je lahko element podatkovne množice. Tipičen primer gručenja na podlagi centroidov je k-voditeljev (angl. k-means). Algoritem si v začetni fazi izbere k centralnih vektorjev, kjer je k vnaprej podan vhodni parameter.

Cilj algoritma je minimizacija funkcije $E(C)$, ki je vsota kvadrata evklidskih razdalj [2] med vsako točko znotraj gruče in centralnim vektorjem.

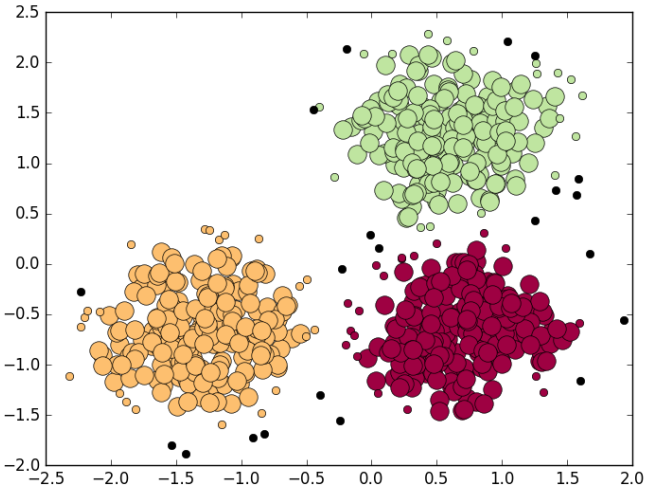
$$E(C) = \sum_{j=1}^{k} \sum_{x_i \in C_j} \|x_i - c_j\|^2 , \tag{2.6}$$

kjer C predstavlja rezultat gručenja, c_j j-ti centralni vektor in x_i i-ti element [2]. Algoritem iterativno optimizira E(C) do ustavitvenega pogoja (na primer, ko se gručenja ne spreminjajo več). Poznamo dve različici k-voditeljev iterativne optimizacije. Prva je sestavljena iz dveh korakov: (1) vsem točkam se določi gruča glede na najbližji centralni vektor in (2) nove centralne vektorje izračunamo s povprečenjem vseh točk, ki spadajo v določeno gručo. Druga različica pa spremeni centralne vektorje samo, če sprememba doprinese k izboljšavi $E(C)$ [2]. Optimizacijski proces pa ne najde vedno najboljše rešitve. Ob neustrezni izbiri začetnih centralnih vektorjev lahko proces konvergira v lokalni optimum, zato je za zagotovitev zanesljivih rezultatov smiselno večkratno izvajanje algoritma. Algoritem je občutljiv na osamelce in ima težave s stabilnostjo, saj lahko majhne spremembe v podatkih povzročijo popolnoma drugačen rezultat gručenja.

Verjetnostno ali porazdelitveno gručenje je vrsta algoritmov, ki uporabljajo statistične porazdelitvene modele. Gruče so definirane kot skupine elementov, ki z največjo verjetnostjo spadajo v isto verjetnostno porazdelitev [2]. Primer algoritma za verjetnostno gručenje je LDA [16].

Algoritmi gručenja na podlagi gostote tipično ne potrebujejo vhodnega parametra k in se dobro soočajo s problemom osamelcev. Najbolj znan primer je algoritem DBSCAN. Kot vhodne parametre prejme podatkovno zbirko, $minPts$, ki določa najmanjše število točk, ki še lahko sestavljajo gručo in ϵ, ki določa okolico. V prvi fazi se vse točke označi kot jedrne ali

[2] Evklidska razdalja je običajna razdalja v evklidski geometriji. Evklidska razdalja med dvema točkama v $\mathbb{R}^n$ se izračuna z $d(a,b) = \|a-b\| = \sqrt{(a_1-b_1)^2 + (a_2-b_2)^2 ... + (a_n-b_n)^2}$



Slika 2.2: Dvodimenzionalna vizualizacija gručenja z algoritmom DBSCAN. Skupine rumenih, zelenih in rdečih točk predstavljajo tri gruče, črne točke pa osamelce.

mejne ali osamelce. Točka je jedrna, če je v njeni okolici ϵ vsaj $minPts$ točk. Točka je mejna, če je v njeni okolici manj kot $minPts$ točk, a je vsaj ena izmed njih jedrna. Točka je osamelec, če je v njeni okolici manj kot $minPts$ točk in nobena ni jedrna. V drugi fazi vsako jedrno in mejno točko umestimo v gručo. Najprej izberemo neoznačeno jedrno točko in umestimo vse točke v njeni okolici v isto gručo. Za vsako jedrno točko, ki je v okolici, se postopek iterativno ponovi z dodajanjem točk v isto gručo. Ko se iterativni postopek konča, se za naslednjo gručo izbere naslednjo neoznačeno jedrno točko. Proces se ponavlja, dokler niso vse točke razen osamelcev umeščene v gruče. DBSCAN lahko najde gruče poljubnih oblik. Slabost algoritma je potreba po ustreznih vhodnih parametrih ϵ in $minPts$ za vsako podatkovno zbirko [2].

Poleg opisanih metod so se razvile tudi druge metode gručenja, ki ne spadajo v nobeno od zgoraj naštetih kategorij. Te so npr. spektralno gručenje, BIRCH, umetne nevronske mreže za nenadzorovano učenje, gručenje na pod-

lagi omejitev (angl. constraint based clustering) in evolucijske metode gručenja [2].

Spektralno gručenje temelji na uporabi lastnih vrednosti matrike podobnosti primerov za zmanjšanje dimenzionalnosti pred postopkom gručenja. Kot vhodni parameter prejme tudi število gruč [31].

BIRCH je metoda gručenja primerna za visokodimenzionalne podatkovne zbirke. Tako kot DBSCAN lahko prepozna osamelce [49].

2.3 Kriteriji za oceno kakovosti gručenja

Pri nadzorovanih metodah strojnega učenja, kjer so oznake razredov podane, se uspešnost modela ocenjuje s preprostimi tehnikami, kot so klasifikacijska točnost, točnost in priklic. Kljub temu, da pri gručenju oznak razredov ni, je za izbiro pravih vhodnih parametrov, metod in splošne ocene uspešnosti pomembna uporaba kriterijev za oceno gručenja. Gručenja ocenjujemo z notranjimi in zunanjimi kriteriji.

2.3.1 Notranji kriteriji za oceno kakovosti gručenja

Notranji kriteriji za oceno kakovosti gručenja temeljijo zgolj na ocenjevanju s pomočjo podatkov, ki so bili uporabljeni v procesu gručenja. Slabost notranjih kriterijev za oceno gručenja je, da dobre ocene gručenja nujno ne odražajo smiselnosti in uporabnosti pri aplikativnih nalogah. Prav tako so nekatere mere pristranske do uporabe določenih metrik. Na primer, k-voditeljev lahko najde samo konveksne oblike gruč in mere, ki predpostavljajo konveksne oblike gruč, bodo tako bolje ocenile k-voditeljev kot DBSCAN, ki pa išče gruče poljubnih oblik [14]. Splošna ideja gručenja je najti takšne gruče, da bodo elementi znotraj gruč med seboj bolj podobni, kot elementi zunaj gruč. Tako notranje mere tipično temeljijo na dveh kriterijih [37]:

I. Kompaktnost meri, kako podobni so si elementi znotraj gruč. Veliko kriterijev ocenjuje kompaktnost glede na varianco. Manjša varianca odraža večjo kompaktnost gruč. Ostali kriteriji ocenjujejo kompaktnost glede na

razdaljo. Na primer, maksimalna in povprečna razdalja med pari elementov znotraj gruč ali maksimalna in povprečna razdalja med elementi in centri gruč. Kompaktnost ni zadosten pogoj za celovito oceno kakovosti gručenja, saj so lahko gruče med seboj zelo blizu ali pa se, v primeru gručenja s prekrivanjem, popolnoma prekrivajo brez negativnega vpliva na kompaktnost. Tako se poleg kompaktnosti uporablja še drug kriterij.

II. Ločenost meri, kako ločene so gruče od ostalih gruč. Pogosta mera ločenosti je razdalja med pari centralnih vektorjev gruč ali minimalna razdalja parov elementov iz različnih gruč. Prav tako se uporabljajo mere, ki temeljijo na gostoti porazdelitve primerov.

Notranji kriteriji se tipično uporabljajo tako, da se gručenje večkrat izvede z različnimi metodami in različnimi vhodnimi parametri, nato vsako gručenje ocenimo z notranjimi kriteriji. Najboljša ocena odraža najustreznejše vhodne parametre in najustreznejšo metodo gručenja [37]. Glavne metode notranjih kriterijev gručenja so naslednje [37, 28]:

- **Calinski-Harabasz indeks**

Calinski-Harabasz indeks ovrednoti uspešnost gručenja glede na razpršenost elementov znotraj gruč (ki vpliva negativno) in razpršenost elementov med gručami (ki vpliva pozitivno). Izračuna se z naslednjo formulo:

$$CH = \frac{sled(S_B)}{sled(S_W)} * \frac{n_p - 1}{n_p - k}, \tag{2.7}$$

kjer je S_B matrika razpršenosti med gručami in S_W matrika razpršenosti znotraj gruč. n_p predstavlja število označenih elementov in k število gruč. Sled je vsota diagonalnih elementov kvadratne matrike. Večja vrednost CH odraža boljše gručenje.

- **Davies-Bouldin indeks**

Davies-Bouldin indeks se izračuna na podlagi povprečne razdalje med elementi znotraj gruče in centralnim vektorjem.

$$DB = \frac{1}{k}\sum_{i=1}^{k}\max_{j\neq i}\left(\frac{\sigma_i + \sigma_j}{d(c_i, c_j)}\right), \tag{2.8}$$

kjer c_i predstavlja centralni vektor i-te gruče, σ_i predstavlja povprečno razdaljo med centralnim vektorjem i-te gruče in pripadajočimi elementi ter $d(c_i, c_j)$ razdaljo med centralnima vektorjema $i - te$ in j-te gruče. Manjša vrednost odraža boljše gručenje.

- **Dunnov indeks**

Dunnov indeks (J. C. Dunn 1974) išče goste in dobro ločene gruče. Definiran je kot razmerje med minimalno povprečno razdaljo med gručami $d(C_i, C_j)$ in maksimalno povprečno razdaljo elementov znotraj gruč $d'(C_k)$.

$$DI = \frac{\min\limits_{1\leqslant i<j\leqslant n} d(C_i, C_j)}{\max\limits_{1\leqslant k\leqslant n} d'(C_k)} \tag{2.9}$$

Vrednosti $d'(C_k)$ in $d(C_i, C_j)$ sta lahko izračunani glede na povprečne medsebojne razdalje parov elementov ali povprečne medsebojne razdalje elementov in centralnih vektorjev. Večja DI vrednost odraža boljše gručenje.

- **Silhueta**

Silhueta podobno kot Dunnov indeks deluje na principu medgručnih in znotrajgručnih razdalj parov elementov, a izračuna oceno za vsak element posebej. Tako so glede na silhueto elementi z visoko oceno dobro uvrščeni, elementi z nizko oceno pa so z veliko verjetnostjo osamelci. Silhueta za i-ti element se izračuna z:

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}, \tag{2.10}$$

kjer je $a(i)$ mera povprečne razdalje med i-tim elementom in ostalimi elementi znotraj iste gruče in $b(i)$ minimalna povprečna razdalja med i-tim elementom in elementi ostalih gruč. V najboljšem primeru (kjer je v gruči samo en unikaten element) je $a(i)$ enak 0 in vrednost silhuete enaka 1. V najslabšem primeru, kjer je $b(i)$ enak 0, je pa vrednost silhuete enaka -1. Tako je silhueta kriterij za katerega velja $-1 \leq s(i) \leq 1$. Ker $s(i)$ poda oceno le za en element, je za splošno oceno gručenja potrebno izračunati $s(i)$ za vsak element in dobljene vrednosti povprečiti. Silhueta se uporablja tudi za oceno optimalnega števila k.

2.3.2 Zunanji kriteriji

Zunanji kriteriji gručenje ocenijo na podlagi informacij, ki niso bile uporabljene pri gručenju. Te lahko temeljijo na predhodnem znanju o podatkih v obliki oznak razredov ali so pa v primeru večslojnega gručenja rezultat gručenja na drugem sloju.

Ideja večine pristopov temelji na štetju soglasij in nesoglasij med pari vzorcev glede na to, ali določen par spada ali ne spada v isto skupino pri gručenju in glede na zunanje informacije. Rezultat gručenja označimo s C, zunanje informacije, ki si jih za lažje razumevanje lahko predstavljamo kot oznake razreda, označimo s C'. Ujemanje parov gručenj si lahko predstavljamo z naslednjo kontingenčno tabelo 2.1 [22]:

Tabela 2.1: Kontingenčna tabela ujemanja C' in C, kjer so vse možne kombinacije parov elementov razdeljene na pare, ki spadajo v isto gručo P, in pare elementov, ki spadajo v različne gruče N.

C / C'	P	N
P'	RP	LN
N'	LP	RN

Takšna kontingenčna tabela se pogosto uporablja pri dvorazrednih klasifi-

kacijskih problemih. Tam se tipično uporabljajo oznake razredov 1 (pozitivni primeri) in 0 (negativni primeri). Tako so pri gručenju P pari elementov, ki so uvrščeni v isto gručo in N pari elementov, ki spadajo v različne gruče.

RP (resnični pozitivni primeri) predstavlja število parov, ki spadajo v isto gručo tako v C kot v C'

LN (lažni negativni primeri) predstavlja število parov, ki v C' spadajo v isto gručo, v C pa v različne gruče.

LP (lažni pozitivni primeri) predstavlja število parov, ki v C spadajo v isto gručo, v C' pa v različne gruče.

RN (resnični pozitivni primeri) predstavlja število parov, ki tako v C kot tudi v C' spadajo v različne gruče.

Na podlagi kontingenčne tabele 2.1 lahko definiramo točnost in priklic[34] :

$$\begin{aligned} Točnost &= \frac{RP}{RP + LP}, \\ Priklic &= \frac{RP}{RP + LN}. \end{aligned} \tag{2.11}$$

Pri binarni klasifikaciji je točnost delež pozitivno prepoznanih primerov, ki so zares pozitivni in priklic delež zares pozitivnih primerov, ki so bili prepoznani kot pozitivni. V primeru primerjave dveh gručenj je ta razlaga manj razumljiva, a se meri vseeno uporabljata pri zunanjih kriterijih za oceno gručenj, kot je Fowlkes–Mallows indeks in mera F[37].

- **Fowlkes–Mallows indeks**

Fowlkes–Mallows indeks (FM) oceni podobnost med dvema gručenjema elementov. V članku [15] se Fowlkes–Mallows indeks uporablja za primerjavo večih hierarhičnih gručenj na istih podatkih. Izračuna se s formulo:

$$FM(C, C') = \sqrt{\frac{RP}{RP + LP} \cdot \frac{RP}{RP + LN}}, \tag{2.12}$$

kjer večja FM vrednost predstavlja večjo podobnost gručenj. Enačba predstavlja koren zmnožka točnosti in priklica. Tako si mero lahko razlagamo kot

geometrično sredino med točnostjo in priklicom [45].

- **Randov indeks**

Randov indeks (William M. Rand 1971) gručenje primerja z zunanjo informacijo z uporabo formule [45, 22]:

$$RI(C, C') = \frac{RP + RN}{RP + LP + LN + RN}, \tag{2.13}$$

kjer $(RP + LP + LN + RN)$ predstavlja število vseh možnih parov elementov podatkovne zbirke. Vrne vrednost med 0 in 1, kjer 1 pomeni, da se C popolnoma sovpada s C' in 0, da sta C in C' popolnoma ločena. Slabost Randovega indeksa je, da ne upošteva pričakovane vrednosti ujemajočih se parov. Tako naključna gručenja v splošnem vrnejo vrednost večjo od 0.

- **Popravljen Randov indeks**

Popravljen Randov indeks (angl. Adjusted Rand Index - ARI) [45, 22] je razširitev zgornje metode, ki upošteva pričakovano vrednost Randovega indeksa. Vrne vrednost med -1 in 1, kjer večje število odraža boljše ujemanje med C in C'. Za razumevanje delovanja je potrebna razširitev kontingenčne tabele 2.1.

Tabela 2.2 prikazuje ujemanje elementov gruč v C in C', kjer n_{lk} predstavlja število elementov, ki so skupni l-ti gruči iz C' in k-ti gruči iz C. a_l je seštevek ujemanj elementov med C'_l in gručami $C_{1,2..k}$, kar je enako številu vseh elementov v gruči C'_l. Prav tako b_k predstavlja število elementov v gruči C_k. Izračuna se z naslednjo formulo:

$$ARI(C, C') = \frac{\sum_{ij} \binom{n_{ij}}{2} - [\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}]/\binom{n}{2}}{\frac{1}{2}[\sum_i \binom{a_i}{2} + \sum_j \binom{b_j}{2}] - [\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}]/\binom{n}{2}} \tag{2.14}$$

Tako ARI za naključna gručenja vrne vrednost blizu 0. V primeru, da je Randov indeks manjši od pričakovane vrednosti, lahko vrne tudi število

Tabela 2.2: Kontingenčna tabela ujemanja elementov med posameznimi gručami v C in C'

C / C'	C_1	C_2	...	C_k	vsota
C'_1	n_{11}	n_{12}	...	n_{1k}	a_1
C'_2	n_{21}	n_{22}	...	n_{2k}	a_2
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
C'_l	n_{l1}	n_{l2}	...	n_{lk}	a_l
vsota	b_1	b_2	...	b_k	

manjše od 0. Ker je tabela 2.2 simetrična, je tudi ARI simetrična mera $ARI(C, C') = ARI(C', C)$.

- **Jaccardov indeks**

Jaccardov indeks je preprosta mera za izračun podobnosti dveh razdelitev [22]. Vrne vrednost med 0 in 1 in se izračuna s sledečo formulo:

$$J(C, C') = \frac{RP}{RP + LP + LN}. \tag{2.15}$$

Pogosto se uporablja v geologiji in ekologiji za primerjavo podobnosti različnih vrst [45].

- **Variacija informacije**

Variacija informacije (angl. variation of information - VI) je mera za oceno podobnosti dveh gručenj, ki temelji na medsebojni informaciji [45]:

$$I(C, C') = \sum_{i=1}^{k} \sum_{j=1}^{l} P(i, j) \log_2 \left(\frac{P(i, j)}{P(i)\, P(j)} \right), \tag{2.16}$$

kjer je $P(i, j)$ verjetnost, da element hkrati spada v i-to gručo C in j-to gručo C'. Izračuna se s formulo [22]:

$$VI(C, C') = H(C) + H(C') - 2I(C, C') \tag{2.17}$$

kjer $H(C)$ predstavlja entropijo razdelitve C. Mera lahko zavzame vrednosti od 0 (popolno ujemajoči se razdelitvi) do $log(N)$ (neujemajoči se razdelitvi), kjer je N število vseh elementov.

- **Normalizirana medsebojna informacija**

Normalizirana medsebojna informacija (angl. normalized mutual information - NMI) oceni podobnost dveh razdelitev elementov s funkcijo [45]:

$$NMI(C, C') = \frac{I(C, C')}{\sqrt{H(C)H(C')}}\,, \tag{2.18}$$

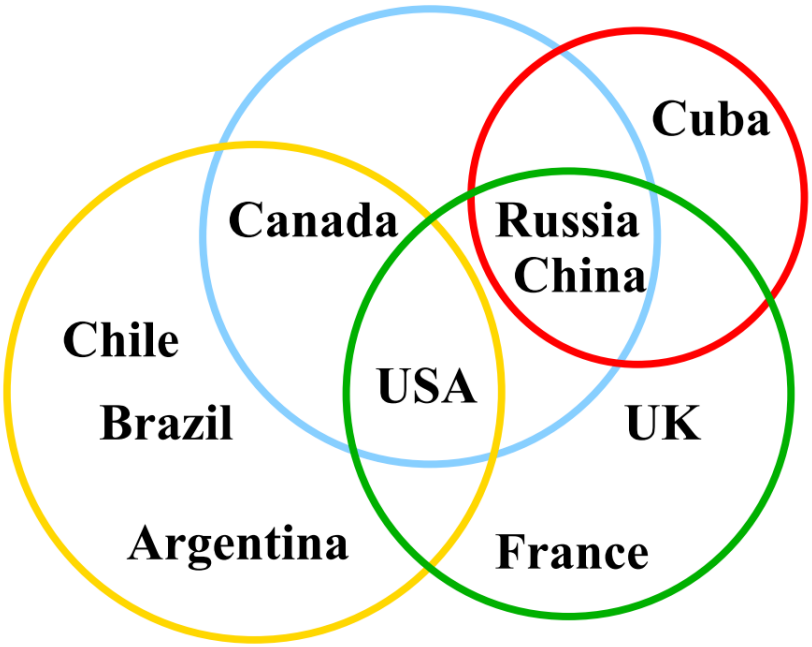
kjer $I(C, C')$ predstavlja medsebojno informacijo med C in C', $H(C)$ pa entropijo C. Mera je podobna variaciji informacije. Ker je medsebojna informacija simetrična mera, je tudi normalizirana medsebojna informacija simetrična.

Ker zunanji kriteriji ne primerjajo le gručenja z oznakami razredov, temveč tudi več gručenj med seboj, predstavljajo ključne metrike uspešnosti pri večslojnem gručenju.

2.4 Večopisno rudarjenje

Večopisno rudarjenje (angl. redescription mining) je novejša tehnika strojnega učenja, ki, tako kot učenje z večimi pogledi, temelji učenju na več skupinah značilk. Cilj večopisnega rudarjenja je najti takšne skupine objektov, ki jih lahko opišemo na več načinov. Temelji na predpostavki, da je rezultat, ki ga potrjujeta dva različna vira podatkov bolj verodostojen, kot rezultat, ki potrjuje en sam vir. Področje je podobno učenju z večimi pogledi. Cilj učenja z večimi pogledi je izgradnja modela za čim uspešnejše gručenje ali klasifikacijo, cilj večopisnega rudarjenja pa je za vsako dobljeno skupino najti opis iz vsake skupine značilk [33].

Slika 2.3 prikazuje enostaven primer večopisnega rudarjenja. Vsak krog objekte (v tem primeru države) razvršča v skupine glede na neko defini-



Slika 2.3: Primer večopisnega rudarjenja (Parida in Ramakrishnan, 2005).

cijo. Na primer, zelena, rdeča, modra in rumena predstavljajo zapored 'države članice varnostnega sveta Združenih narodov', 'države z zgodovino komunizma', 'države z več kot 777 milijonov m^2 kilometrov', 'znane turistične destinacije v južni in severni Ameriki'. Tovrstnim razdelitvam pravimo opisi. Primer večopisne razdelitve se glasi: opis 'države z več kot 777 milijonov m^2 kilometrov zunaj južne in severne Amerike' je ekvivalenten opisu 'države članice varnostnega sveta Združenih narodov, ki imajo zgodovino komunizma'. Opisa zajemata države Rusijo (angl. Russia) in Kitajsko (angl. China). Enkrat sta definirani s presekom množic in enkrat z razliko množic [33].

Uporabna lastnost večopisnega rudarjenja je, da so opisi dobljenih skupin interpretabilni. So tipično v obliki logičnih formul ali odločitvenih dreves, ki

so opisna v razdelku 2.4.1. Z opisa je razvidno, kakšne kombinacije in vrednosti značilk so prispevale k dobljeni razdelitvi. To lahko pri realnih podatkih prispeva k razlagi nastanka skupin in odkrivanju povezav med značilkami skupin. [33]. V nadaljevanju opišemo eno od metod večopisnega rudarjenja.

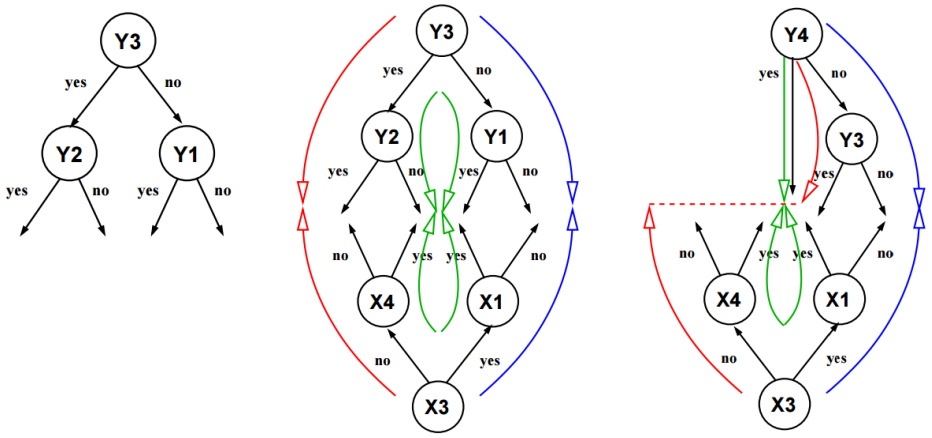
2.4.1 Metoda CARTwheels

Klasifikacijska in regresijska drevesa (angl. Classification and regression trees - CART) so ena najzgodnejših metod strojnega učenja. Rezultat določimo glede na niz pravil, ki temeljijo na vrednostih značilk. Lahko jih predstavimo kot drevo, kjer listi predstavljajo razrede [40]. Ta pravila so človeku razumljiva in so podobna odločitveni logiki strokovnjakov z različnih področij.

Ramakrishnan in sod. (2004) so kot začetniki večopisnega rudarjenja predstavili metodo CARTwheels. Temelji na izgradnji dveh odločitvenih dreves, ki rasteta v obratne smeri in sta združena v listih. Ti predstavljajo razdelitve objektov. Odločitvena drevesa pri različnih skupinah značilk v splošnem pripeljejo do različnih razvrstitev objektov. Cilj metode CARTwheels je izgradnja dveh odločitvenih dreves, ki imata v listih enake razdelitve objektov in se lahko v listih spojita. Ko so rezultati obeh dreves enaki, rečemo, da so razdelitve objektov določene večopisno [36].

Algoritem prejme množico objektov $O = \{o_1, o_2, ..., o_n\}$, ki jih opisujeta dve skupini značilk X, Y. Sprva se zgradi odločitveno drevo na podlagi značilk Y, kot je razvidno z levega dela slike 2.4. Nato se na podlagi X zgradi še drugo drevo, ki se popravi tako, da se rezultati dreves bolj ujemajo (sredinski del slike 2.4). Nato se zgornje drevo zgradi na novo, tako da se čim bolj ujema z rezultati spodnjega drevesa (desni del slike 2.4). Za določanje stopnje ujemanj rezultatov dreves se uporablja Jaccardov indeks, ki je opisan v razdelku 2.3.2. Proces se ponavlja, dokler se rezultati obeh dreves ne ujemajo. Takrat je model večopisen [36].

Večopisno rudarjenje se kljub zanesljivim rezultatom na določenih dome-



Slika 2.4: Delovanje algoritma CARTwheels (Kumar in sod, 2004). $Y_{1,2,3}$ so binarne značilke iz Y. $X_{1,2,3}$ so binarne značilke iz X. Barve puščic predstavljajo ujemajoče se pare.

nah sooča s težavami. Slabost večopisnega rudarjenja je, da je zelo občutljivo na šumne podatke, ki so skoraj vedno prisotni pri realnih podatkih. To lastnost imajo tudi odločitvena drevesa. Že ena šumna značilka lahko povzroči napačne razdelitve objektov, saj v tem primeru metode večopisnega rudarjenja skušajo najti opis objektov, ki temelji na neinformativnih podatkih. To negativno vpliva tudi na izgradnjo drugega drevesa in nazadnje na rezultat [19].

2.5 Večslojno gručenje

Večslojno gručenje je tehnika, ki se prav tako kot večopisno rudarjenje, uporablja za odkrivanje povezav med večimi skupinami značilk (pogledi), a je manj občutljiva na šumne podatke. Definirana je kot kombinacija večopisnega rudarjenja in učenja z večimi pogledi. Temelji na učenju na večih skupinah značilk, ki jih imenujemo sloji. Uspešnost večslojnega gručenja temelji na homogenosti gruč v dveh ali večih slojih [18, 19].

Gamberger in sod. (2015) opišejo primer metode večslojnega gručenja, ki temelji na oceni raznolikosti gručenj (angl. Clustering Related Variability - CRV). Za lažje razumevanje sprva predstavimo enoslojno izvedbo algoritma. Gručenje izvajamo iterativno od spodaj navzgor (angl. bottom up), tako da sprva vsak element spada v svojo gručo. Nato združimo tisti par gruč, ki povzroči največje izboljšanje ocene CRV. To naredimo tako, da izračunamo CRV oceno za vsak par gruč. Združevanje ponavljamo, dokler združitev nobenega para ne povzroči izboljšanja CRV ocene. V procesu posledično najdemo tudi primerno število gruč.

Večslojna izvedba algoritma je enaka, le da CRV oceno izračunamo za vsak sloj posebej in združitev izvedemo le, če se ocena izboljša v vseh slojih. Omenjena metoda je bila uspešno uporabljena na podatkih o Alzheimerjevi bolezni z namenom iskanja povezav med biološkimi in kliničnimi značilkami [18].

2.6 Metode za izbiro značilk

Učni primeri v strojnem učenju so tipično opisani z večimi lastnostmi, ki jih imenujemo značilke. Nekatere podatkovne zbirke vsebujejo veliko število značilk. V praksi se izkaže, da k izgradnji uspešnega modela tipično prispeva le neka podmnožica značilk, ostale pa so neinformativne. Te ne le, da ne doprinesejo k uspešnemu učenju, temveč lahko uspešnost modela celo pokvarijo. Primer algoritma, ki je občutljiv na šumne podatke, je k-najbližjih sosedov, saj temelji na razdaljah med primeri, na katere neinformativne značilke vplivajo prav tako kot informativne. Na šumne podatke je občutljiv tudi algoritem k-means, ki prav tako temelji na razdaljah med primeri.

Metode za izbiro značilk z različnimi tehnikami ocenijo pomembnost značilk in omogočajo odstranjevanje nepomembnih. Zaradi zmanjševanja dimenzionalnosti to v splošnem pohitri proces učenja, izboljša klasifikacijsko točnost in omogoča lažje razumevanje rezultatov [6]. Primer področja, kjer je izbira značilk nujna, so raziskave izražanja genov. Tam imamo stotine

značilk, ki so med seboj močno korelirane. Metode izbire značilk pogosto zamenjamo s sorodnimi tehnikami za zmanjševanje dimenzionalnosti, kot je metoda glavnih osi, ki pa ne izbira značilk, temveč generira nove. Izčrpno preiskovanje, ki bi preiskalo vse možne podmnožice značilk, ne pride v poštev, saj število možnih podmnožic narašča eksponentno. Za podatkovno zbirko z N značilkami bi morali preiskati 2^N podmnožic. Tako se pojavi potreba po tehnikah, ki značilke izbirajo učinkovitejše [6].

Metode izbire značilk delimo na tri glavne kategorije:

- filtrirne metode,
- ovojne metode (angl. wrapper methods),
- vgrajene metode.

Filtrirne metode, ki za izbiro značilk uporabljajo učne podatke (in pripadajoče razrede), so neodvisne od učnega modela. Vsako značilko ocenijo z določeno vrednostjo. S sortiranjem značilk po vrednosti ocen izberemo poljubno veliko podmnožico najboljših značilk. Slabost filtrirnih metod je, da jih večina zaradi odsotnosti modela ne upošteva medsebojnih odvisnosti med značilkami. Izjema je algoritem ReliefF [38], ki uspešno prepozna odvisnosti med spremenljivkami.

Ovojne metode iščejo najboljšo podmnožico značilk glede na uspešnost učnega modela. Poznamo sekvenčne in hevristične ovojne metode. Sekvenčne metode začnejo z množico vseh značilk (ali z nobeno) in odstranjujejo (ali dodajajo) značilke glede na to, katera značilka najbolj doprinese k uspešnosti modela. Proces se iterativno ponavlja do lokalnega optimuma uspešnosti učnega modela [6]. Najbolj preprost primer sekvenčne metode, ki v vsaki iteraciji preizkusi odstranjevanje (dodajanje) vsake značilke v množici, ima za N značilk zahtevnost preiskovanja $O(N^2)$. V prvi iteraciji preverimo uspešnost modela za odstranjevanje vsake značilke, torej preverimo N kombinacij. Ko odstranimo značilko, ki je prispevala k najboljši

oceni, imamo le $N-1$ značilk, in pri preverjanju uspešnosti v naslednji iteraciji preverimo le $N-1$ kombinacij. Končno je število preizkusov enako $N+(N-1)+(N-2)+...+1 = \sum_{k=1}^{N} k = \frac{N(N+1)}{2}$, torej je zahtevnost preiskovanja enaka $O(N^2)$. To je mnogo bolje kot izčrpno preiskovanje, ki ima zahtevnost $O(2^N)$. Hevristične ovojne metode za izbiro podmnožic značilk uporabljajo različne hevristike in, tako kot sekvenčne, na koncu izberejo podmnožico značilk, ki prinese najboljši model.

Pri vgrajenih metodah je proces izbire značilk tipično del učenja modela in tako specifičen za vsak algoritem [6]. V nadaljevanju opišemo nekaj filtrirnih metod za izbiro značilk.

- **Korelacijski koeficient**

Ena najpreprostejših tehnik izbire značik temelji na korelacijskemu kriteriju. Ta značilke oceni na podlagi stopnje korelacije med oznakami razredov in vrednostmi značilk. Značilko x oceni na podlagi korelacije z oznakami razredov z uporabo Pearsonovega korelacijskega koeficienta [25]:

$$r = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}, \tag{2.19}$$

kjer $\bar{x}$ predstavlja povprečno vrednost značilke x in $\bar{y}$ povprečno vrednost oznak razredov. Korelacijski kriteriji lahko odkrijejo le linearne odvisnosti med razredom in značilkami. Primeri so za številske vrednosti razreda in značilk.

- **Medsebojna informacija**

Medsebojna informacija, ki je opisana v poglavju 2.3.2, je še en primer filtrirne metode za izbiro značilk. Za posamezno značilko in pripadajoče oznake razredov izračuna stopnjo skupne informacije, ki določa uporabnost značilke [6].

- **Relief**

Relief je družina algoritmov za izbiro in ocenjevanje značilk. Uporabljajo se tudi pri izgradnji odločitvenih dreves tako v klasifikaciji kot tudi pri regresiji. Uporabljajo se lahko tako za klasifikacijo kot za regresijo. Večina algoritmov za ocenjevanje značilk predpostavlja pogojno neodvisnost med značilkami, ki je opisana v poglavju 2.1.4, zato niso primerni za podatkovne zbirke, kjer med značilkami najdemo veliko interakcije. Družina algoritmov Relief uspešno dela tudi s pogojno odvisnimi značilkami in lahko tudi odvisnosti tipa XOR.

Algoritem Relief prejme n učnih primerov, p značilk in parameter m, ki določi število iteracij. Za vrednosti značilk je pomembno, da so skalirane na enoten interval, npr. $[0 - 1]$, saj algoritem temelji na razdaljah in bi bile razdalje med značilkami sicer neuravnotežene.

Relief (algoritem 1) značilke oceni na podlagi stopnje ločevanja primerov, ki so si blizu. Za naključni primer x_i poiščemo najbližji zadetek H in najbližji pogrešek M. H je najbližji učni primer, ki spada v isti razred kot x_i in M najbližji učni primer, ki spada v drug razred kot x_i. Nato popravimo utež $W[A]$ za vsak atribut A glede na vrednosti x_i, M in H. Ko primerjamo razliko med vrednostmi atributa A za x_i in H (ki spadata v isti razred), želimo, da so razlike čim manjše, saj želimo, da imajo primeri iz istih razredov podobne vrednosti atributov. Pri razdalji vrednosti atributa A za x_i in M (ki spadata v različna razreda), želimo, da so razdalje čim večje in tako čim bolj ločujejo učne primere različnih razredov. Relief za razliko vrednosti primerov iz istih razredov značilke nagrajuje in za razliko vrednosti primerov iz različnih razredov značilke kaznuje. Celoten postopek ponovimo za m primerov.

Rezultat algoritma je vektor uteži W, ki vsebuje ocene značilk. Za razdaljo med vrednostmi dveh numeričnih značilk uporabljamo funkcijo:

$$diff(A, x_i, x_j) = \frac{|vrednost(A, x_i) - vrednost(A, x_j)|}{max(A) - min(A)}, \tag{2.20}$$

Algoritem 1 Psevdokoda algoritma Relief

```
1: nastavi vse uteži W[A] := 0.0;
2: for i = 1 to m do
3:    izberi naključni primer x_i
4:    najdi najbližji zadetek H in najbližji pogrešek M
5:    for A = 1 to a do
6:       W[A] = W[A] − diff(A, x_i, H)/m + diff(A, x_i, M)/m
7:    end for
8: end for
```

kjer $min(A)$ predstavlja najmanjšo vrednost A in $max(A)$ največjo vrednost A. Za značilke kategoričnih vrednosti velja, da je $diff(A, x_i, x_j)$ enaka 0, če sta vrednosti enaki, in 1, če sta vrednosti različni.

Slabost algoritma Relief je, da deluje le z dvorazrednimi problemi. Prav tako ne zna ravnati z manjkajočimi podatki.

Algoritem 2 Psevdokoda algoritma ReliefF

```
1: nastavi vse uteži W[A] := 0.0;
2: for i = 1 to m do
3:    izberi naključni primer x_i
4:    najdi k najbližjih sosedov H_j
5:    for vsak razred C ≠ razred(x_i) do
6:       najdi k najbližjih pogreškov M_j(C), ki spadajo v razred C
7:    end for
8:    for A = 1 to a do
9:       W[A] = W[A] − Σ_{j=1}^{k} diff(A, x_i, H_j)/(m · k) +
         Σ_{C≠class(x_i)} [P(C)/(1−P(class(x_i)))] Σ_{j=1}^{k} diff(A, x_i, M_j(C))]/(m · k)
10:   end for
11: end for
```

Razširitev algoritma, ki rešuje večrazredne probleme in zna ravnati z manjkajočimi podatki, se imenuje ReliefF [38]. Algoritem 2 prejme dodatni parameter k, ki določa število najbližjih sosedov. Zunanja zanka algoritma se, podobno kot pri algoritem Relief, izvede m-krat. Ključna razlika je, da za x_i poiščemo k najbližjih sosedov H_j istega razreda in k najbližjih sosedov

$M_j(C)$, ki spadajo v druge razrede. Nato na podoben način kot pri algoritmu Relief popravimo uteži značilk $W[A]$ za H_j in $M_j(c)$ ter doprinos vseh zadetkov in pogreškov povprečimo. Doprinos pogreškov utežimo s $P(C)$, ki predstavlja delež primerov, ki spadajo v razred C. Ker pri seštevanju doprinosov pogreškov ne upoštevamo vsote za $razred(x_i)$, moramo $P(C)$ deliti še z $1-P(razred(x_i))$, kar predstavlja delež razredov, ki ne spadajo v $razred(x_i)$.

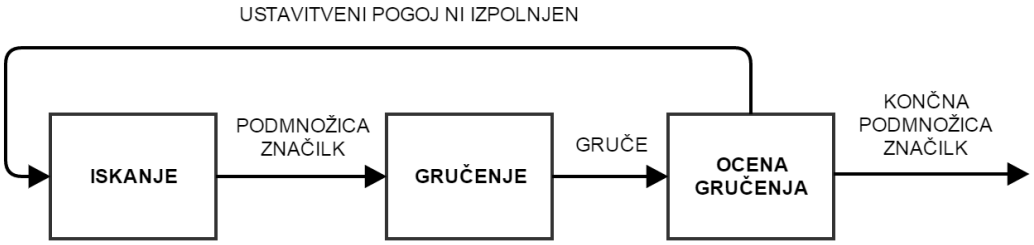
Časovna zahtevnost algoritma ReliefF je $O(m \cdot n \cdot a)$. Najbolj požrešna operacija je iskanje H_j in M_j, saj je za vsak primer R_j potrebno izračunati razdalje do vseh primerov x_i[38].

2.6.1 Izbira značilk pri gručenju

Metode gručenja so izjemno občutljive na šumne podatke. Večina jih predpostavlja, da so vse značilke enako pomembne in tako lahko neinformativne značilke pokvarijo rezultat gručenja. Dodajanje novih dimenzij prispeva k redki (angl. sparse) porazdelitvi podatkov [10].

Izbira značilk pri gručenju ostaja aktualen problem, ki je soroden problemu ocenjevanja kakovosti gručenja. Pri nadzorovanem učenju se lahko pri ovojnih metodah sklicujemo na klasifikacijsko točnost, pri filtrirnih metodah pa na oznake razredov. Pri nenadzorovanem učenju, kjer oznak razredov nimamo, je problem ocenjevanja doprinosa značilk k uspešnemu gručenju težji, saj nimamo dobre definicije, kaj je uspešno gručenje.

Dy in Brodley [13] predlagata sekvenčno metodo z ovojnico za nenadzorovano izbiro značilk. Temelji na iterativnem pristopu od zgoraj navzdol dodajanja značilk glede na kvaliteto gručenja. Uporabljena sta dva kriterija za oceno gručenja; prvi je notranji kriterij, ki temelji na ločenosti razpršenosti (angl. scatter separability), ki je zelo podobna meri silhouette, in drugi je kriterij največje verjetnosti. Algoritem začne s prazno množico in iterativno dodaja značilke glede na oceno gručenja. Za dodajanje vsake značilke izvede gručenje, ki ga nato oceni. V zbirko se doda značilko, ki prispeva k najboljši



Slika 2.5: Izbira značilk z ovojnico pri gručenju.

oceni. Algoritem v zadnjem koraku vrne množico značilk, ki ustvari najboljše ocenjeno gručenje [13]. Dash in sod. [9] predlagajo filtrirno metodo izbire značilk, ki temelji na entropiji.

Eden izmed temeljnih problemov izbire značilk pri gručenju je tudi dejstvo, da imajo lahko nekateri podatki več naravnih in smiselnih gručenj. Na primer, podatkovno zbirko s 1000 značilkami o pacientih lahko razdelimo na dve gruči, ki ločujeta zdrave in bolne paciente, lahko jo razdelimo na deset starostnih skupin ali pa glede na podvrste bolezni pacientov. Visokodimenzionalnost še dodatno prispeva k možnosti, da je smiselnih gručenj več. V razdelku 4.2.2 predlagamo metodo izbire značilk, ki rešuje ta problem z učenjem z večimi pogledi.

Poglavje 3

Podatki o Alzheimerjevi bolezni

Podatki v medicini so pogosto zajeti na več načinov, z večih pogledov. Povezave med značilkami z različnih pogledov zaradi visokodimenzionalnosti in netransparentnosti učnih modelov pogosto niso razumljive. Alzheimerjeva bolezen je ena najpogostejših, še neozdravljivih bolezni. Točni vzroki nastanka še niso pojasnjeni in nobena od ustaljenih tehnik zdravljenja ne ustavi bolezenskega procesa. Ker je bolezen razpoznavna na podlagi kognitivnih sposobnosti, genetske analize, fizioloških znakov, analize proteinov in možganskih slik, je področje primerno za učenje z večimi pogledi.

Za lažje razumevanje empirične evalvacije, ki je opisana v poglavju 5, poglavje predstavi področje bolezni in projekt ADNI, namenjen izboljšanju, odkrivanju in zdravljenju bolezni.

3.1 Alzheimerjeva bolezen

Demenca ali senilnost je družina bolezni možganov, ki vplivajo na kognitivne sposobnosti pacienta. Alzheimerjeva bolezen je najpogostejši tip demence, ki je še neozdravljiv. Poznamo tri oblike bolezni: blaga, zmerna in težka. Znaki blage bolezni so izguba orientacije, poslabšanje kratkoročnega spomina in poslabšanje občutka za čas. Znaki zmerne oblike so poslabšanje kratkoročnega in dolgoročnega spomina, spremembe v obnašanju in izguba

sposobnosti učenja. Znaki težke oblike bolezni so izguba zmožnosti komuniciranja, napadi (angl. seizure), izguba telesne teže, izguba sposobnosti požiranja, odvajanja in dihanja. Ta poslabšanja pogosto povzročijo inhalacijo hrane ali fizične poškodbe ob poskusih premikanja. Težka oblika se skoraj vedno konča s smrtjo [5].

Vzrok bolezni še ni popolnoma razumljiv. Okoli 70% vzrokov naj bi bilo genetskih, ostali faktorji pa so fizične poškodbe glave, depresija in povišan krvni tlak. Diagnoza se sprva postavi na podlagi zgodovine bolezni, kognitivnega testiranja, slikanja možganov, krvnih testov in ostalih kliničnih in bioloških testov. Ti sprva potrdijo, ali gre za tip demence ali za sorodne pojave, kot je naravno staranje. V drugi fazi diagnostika skuša določiti tip demence. Glavne oblike demence so Alzheimerjeva bolezen, demenca z Lewyjevimi telesci in vaskularna demenca. Določanje tipa bolezni ostaja aktualen problem, saj so meje med posameznimi tipi demence zabrisane [5].

3.2 Podatkovna zbirka ADNI

ADNI (angl. Alzheimer's Disease Neuroimaging Initiative) je mednarodni projekt, ki nudi podatke za raziskovanje Alzheimerjeve bolezni. Podatkovno zbirko sestavljajo CT in PET slike možganov, analize genov, analize cerebrospinalnega likvorja, kognitivni in krvni testi.

V raziskavi je sodelovalo 822 pacientov, ki so opravili več obiskov v različnih fazah bolezni. Ob vsakem obisku so pacientom postavili diagnozo. Diagnozo sestavljajo tri kategorije:

- normalno stanje (229 pacientov),
- blaga kognitivna motnja - BKM (405 pacientov),
- demenca (188 pacientov).

V kategorijo normalno stanje so bili uvrščeni pacienti brez poslabšanih kognitivnih sposobnosti ali pacienti z naravno poslabšanimi kognitivnimi sposobnostmi zaradi staranja. V kategorijo blaga kognitivna motnja so bili

uvrščeni pacienti z opazno poslabšanimi kognitivnimi sposobnostmi, ki pa še niso tako kritične, da bi onemogočale normalne delovne aktivnosti. V kategorijo demenca so bili uvrščeni pacienti pacienti s kritično oslabljenimi kognitivnimi sposobnostmi [30].

Celotno podatkovno zbirko sestavlja več kot 150 manjših podatkovnih zbirk, kjer je vsaka zbirka rezultat posamezne raziskave. Vsak pacient je v povprečju opravil le nekaj raziskav, prav noben pacient ni opravil vseh. Tako je izgradnja enotne tabele, ki bi vključevala vse paciente in vse raziskave nesmiselna, saj bi bila takšna tabela zelo redko napolnjena, značilke bi imele več kot 90% manjkajočih vrednosti. Postopek izgradnje podatkovne zbirke iz ADNI podatkov je opisan v razdelku 5.3.1.

Poglavje 4

Metodologija večličnega gručenja za razlago gruč

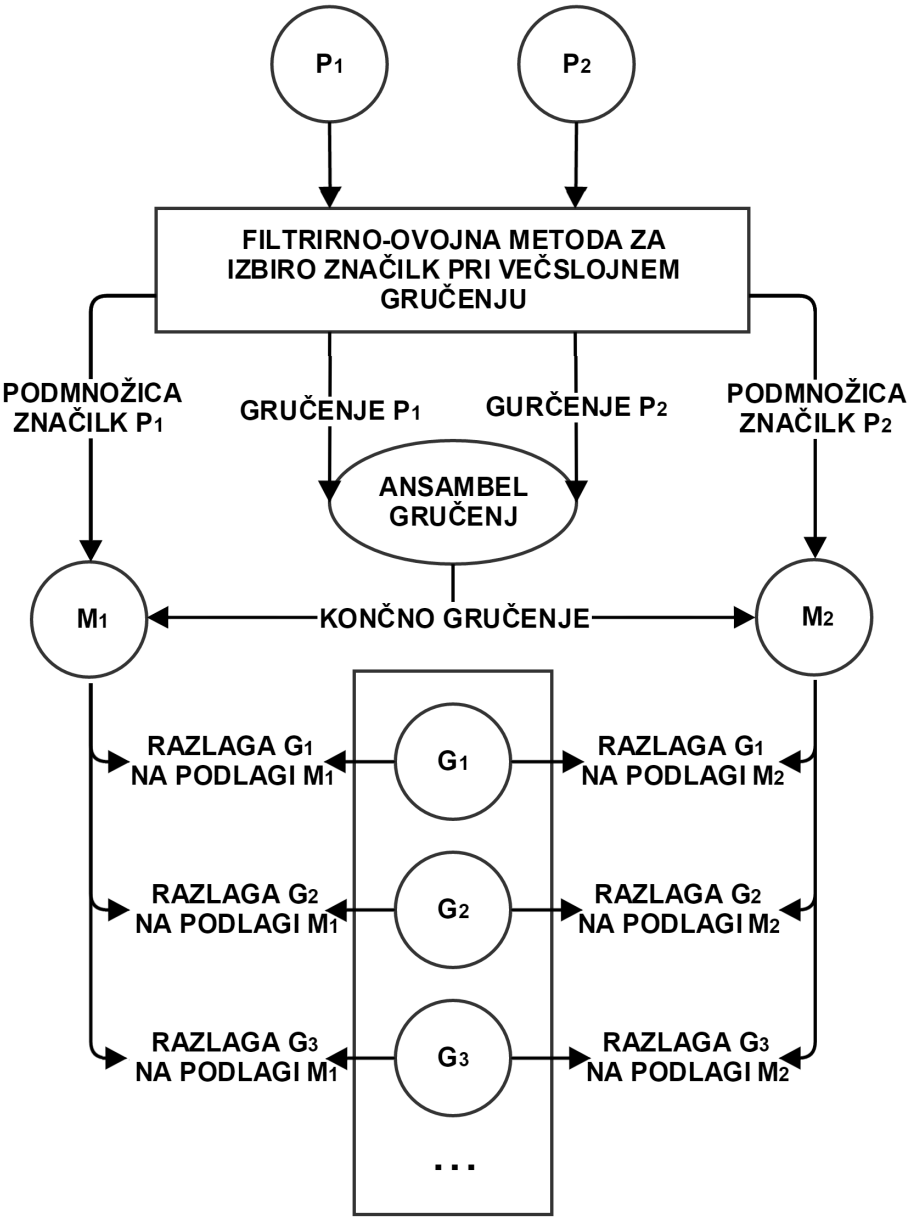
Cilj izbire značilk pri nadzorovanem učenju je preprost. Želimo izbrati takšne značilke, ki bodo izboljšale klasifikacijsko točnost. Pri gručenju želimo izbrati takšne značilke, ki omogočajo čim uspešnejše gručenje. Tovrstne metode bi lahko uporabljali tudi pri večslojnem gručenju, kjer je izbira značilk manj raziskano področje. Tako bi izboljšali uspešnost gručenja na posameznih slojih, a to ne bi nujno prispevalo k ujemanju gručenj, saj imajo lahko podatkovne zbirke več naravnih gručenj. Tak pristop bi prispeval k nastanku več uspešnih, a potencialno neujemajočih se gručenj. V tem poglavju predlagamo metodo izbire značilk pri večslojnem gručenju, ki izbira značilke tudi glede na stopnjo ujemanja gručenj med sloji.

Prvi podrazdelek predstavi splošno idejo metodologije. Drugi predstavi predlagano metodo izbire značilk pri večslojnem gručenju. Tretji podrazdelek vsebuje hiter pregled uporabljenih ansambelskih tehnik. Četrti podrazdelek predstavi pomen in delovanje metod razlage prediktorjev. Zaradi preprostosti razlage primeri temeljijo na učenju iz dveh pogledov.

4.1 Ideja večličnega gručenja z razlago

Predlagana metodologija služi kot metoda večslojnega gručenja, ki uspešno obvladuje tudi visokodimenzionalne podatke. Rezultat je gručenje, kjer je vsaka gruča opisana z značilkami vsakega pogleda posebej na človeku razumljiv način. Metodo sestavljajo trije sklopi. Prvi je izbira značilk pri večslojnem gručenju, ki temelji na spremenjeni izvedbi algoritma ReliefF. Rezultat prvega sklopa so podmnožice značilk, ki prispevajo k dobremu ujemanju gručenj med sloji. Drugi sklop je ansambelska tehnika, ki združuje gručenja v enoten rezultat. Tretji sklop skuša nastale gruče razložiti z vsakega pogleda posebej z uporabo tehnike razlage prediktorjev. Rezultat so gruče opisane na več načinov z značilkami, ki prispevajo k ujemanju gručenj. Ti opisi predstavljajo človeku razumljive povezave med skupinami značilk in omogočajo lažje razumevanje nastanka gruč. Prav tako lahko prva dva sklopa predlagane metodologije uporabljamo kot samostojno tehniko gručenja. V razdelku 5 pokažemo, da lahko s tehniko dosežemo boljše rezultate kot z ostalimi tehnikami gručenja z večimi pogledi.

Diagram 4.1 prikazuje delovanje predlagane metodologije. Na vhodu prejme podatkovno zbirko, ki je opisana na dva načina, s pogledov P_1 in P_2. Podana sta kot vhodna parametra metode za izbiro značilk pri večslojnem gručenju. Rezultat metode sta dve podmnožici značilk, s katerimi dobimo najbolj ujemajoči se gručenji. Ker gre tudi za ovojno metodo, se na vsakem koraku izvaja tudi gručenje. Tako dobimo poleg podmnožic značilk tudi dve najbolj ujemajoči se gručenji. Z ansamblom ustvarimo enotno in bolj stabilno gručenje, ki pa ni bistveno drugačno od vhodnih gručenj, saj rezultat ansambla temelji na ujemanju vhodnih gručenj. M_1 in M_2 predstavljata učna modela za razlago gruč. Končne gruče so na sliki predstavljene z G_1, G_2, G_3.... Entitete označene z *Razlaga...* v spodnjem delu slike so človeku razumljive razlage gruč pridobljene z metodami razlage prediktorjev. Vsaka gruča je razložena posebej s podmnožico značilk P_1 in posebej s podmnožico značilk P_2.



Slika 4.1: Diagram izvajanja predlagane metodologije.

4.2 Izbira značilk

Izbira značilk predstavlja ključen del predlagane metodologije, saj temelji na novi različici ReliefFa za učenje z večimi pogledi. Le-ta deluje na principu konsenza. Gre za ovojno metodo, kar pomeni, da poleg učnih podatkov prejme tudi algoritem gručenja. Namen metode je za podatkovno zbirko, ki je opisana z dvema pogledoma P_1 in P_2, najti podmnožici značilk $S_1 \subseteq P_1$ in $S_2 \subseteq P_2$, na katerih lahko zgradimo najbolj ujemajoči se gručenji. Stopnjo ujemanja ocenimo na podlagi zunanjih kriterijev za oceno gručenja, ki so opisani v poglavju 2.3.2. Takšne ujemajoče se podmnožice značilk so primerne za gradnjo prediktorjev za dano gručenje, čemur sledi smiselna razlaga gruč z večih pogledov.

Metodologija lahko služi tudi kot samostojna tehnika gručenja. Za različne poglede lahko uporabljamo različne tehnike gručenja, kjer izbrana tehnika gručenja ustreza statističnim lastnostim pogleda. Nazadnje gručenja združimo s pomočjo ansambelskih tehnik.

Algoritem 3 opiše delovanje metode za izbiro značilk. Vhodni parametri so algoritem gručenja $C()$ in podatkovna zbirka značilk, ki je razdeljena na pogleda P_1 in P_2. Deluje na principu od zgoraj navzdol, kar pomeni, da iskanje začnemo z vsemi značilkami $S_1 = P_1$ in $S_2 = P_2$. Z uporabo gručenja $C()$ za obe podmnožici izvedemo gručenji $G1 = C(S_1)$ in $G2 = C(S_2)$. Z zunanjim kriterijem za oceno gručenj ocenimo ujemanje gručenj $G1$ in $G2$. Če je ujemanje dosedaj najboljše, si S_1, S_2, G_1 in G_2 shranimo kot najboljši rezultat R. Nato z uporabo tehnike za izbiro značilk z večimi pogledi mvReliefF (Multi View ReliefF) ocenimo značilke S_1 in S_1. Najslabše ocenjeno značilko odstranimo iz S_1 ali S_2 in postopek ponavljamo, dokler so v vsakem pogledu vsaj po dve značilki (izognemo se trivialnemu rezultatu, kjer bi končali z le po eno značilko vsakega pogleda). Kot rezultat vrnemo R.

Algoritem 3 Psevdokoda predlagane metode izbire značilk

```
1: //parametri algoritma so P_1, P_2 in C()
2: bestScore = −1, R = {};
3: S_1 = P_1, S_2 = P_2;
4: while |S_1| > 1 && |S_2| > 1 do
5:    G_1 = C(S_1);
6:    G_2 = C(S_2);
7:    score = scoreF(G_1, G_2); //izmerimo ujemanje gručenj
8:    if score > bestScore then
9:       bestScore = score;
10:      R = {S_1, S_2, G_1, G_2}; //shranimo najboljši podmnožici in gručenji
11:   end if
12:   W_1, W_2 = mvReliefF(S_1, S_2, G_1, G_2); //ocenimo značilke
13:   if min(W_1) < min(W_2) then
14:      S_1 = {A ∈ S_1 | A ≠ znacilka(S_1, index(min(W_1)))}; //odstranimo najslabše
         ocenjeno značilko iz S_1
15:   else
16:      S_2 = {A ∈ S_2 | A ≠ znacilka(S_2, index(min(W_2)))}; //odstranimo najslabše
         ocenjeno značilko iz S_2
17:   end if
18: end while
19: return R;
```

$|S_1|$ predstavlja število značilk v podmnožici značilk S_1. $ScoreF$ predstavlja zunanji kriterij za oceno gručenj in $znacilka(S_i, j)$ predstavlja j-to značilko podmnožico S_1.

4.2.1 Ocena gručenj

Za oceno gručenj lahko uporabimo poljubno zunanjo mero za oceno gručenj, opisano v poglavju 2.3.2. Mera ARI vrne vrednosti med -1 in 1, kjer 1 predstavlja popolnoma ujemajoči se gručenji, -1 predstavlja popolnoma neujemajoči se gručenji, 0 pa pričakovano ujemajoči se gručenji pri naključnem gručenju. Informacija, da je gručenje boljše od naključnega, je pomembna za preverjanje zadostnosti in kompatibilnosti pogledov, zato v nadaljevanju uporabljamo mero ARI. Za primerjavo z rezultati objavljenimi v drugih

člankih, smo uporabljali tudi mero NMI.

4.2.2 Multi View ReliefF

Relief je družina algoritmov za izbiro značilk pri klasifikacijskih problemih, ki zna upravljati z značilkami s pogojno odvisnostjo. Poleg konfiguracijskih parametrov k in m prejme kot vhodni parameter učno množico in pripadajoče oznake razredov. Razdelek predstavlja razširitev algoritma na probleme gručenja z večimi pogledi, ki ga imenujemo mvReliefF (Multi View ReliefF).

Vhodni parametri so podmnožica značilk $S_1 \subseteq P_1$ in $S_2 \subseteq P_2$ ter gručenji G_1 (rezultat gručenja z uporabo S_1) in G_2 (rezultat gručenja z uporabo S_2). Podobni so vhodnim parametrom algoritma Relief, le da namesto oznak razredov podamo oznake gruč. Cilj algoritma je poiskati vektorja ocen W_1 in W_2, ki vsebujeta ocene prispevka k ujemanju gručenj med pogledoma.

Predlagamo tri izvedbe algoritma. Razlikujejo se v tem, kako obravnavajo doprinos značilk k ujemanju gručenj.

mvReliefF-md (Multi View ReliefF - Multi Dinstance) se od ReliefFa razlikuje v računanju razdalj med objekti. ReliefF v vsaki iteraciji za naključni primer x_i izračuna k najbližjih zadetkov H_j in pogreškov $M_j(C)$. H_j so primeri, ki spadajo v isti razred kot x_i, in $M_j(C)$ so primeri, ki ne spadajo v razred primera x_i. Za izračun najbližjih sosedov mora ReliefF v vsaki iteraciji izračunati razdaljo med vsakim učnim primerom in x_i. Pri mvReliefF želimo sprva izračunati W_1 (vektor ocen značilk S_1). Klasična izvedba algoritma ReliefF bi za razdalje upoštevala značilke iz S_1. mvReliefF-md pa za izračun razdalj upošteva značilke obeh pogledov. V primeru evklidske razdalje je razdalja med a-tim in b-tim primerom določena z:

$$dist(a,b) = \sqrt{\sum_{i=1}^{|S_1|}(S_1[a,i] - S_1[b,i])^2 + \sum_{j=1}^{|S_2|}(S_2[a,j] - S_2[b,j])^2}, \qquad (4.1)$$

kjer $S_1[a, i]$ predstavlja vrednost i-te značilke a-tega primera v S_1. $|S_1|$ predstavlja število značilk v S_1. Smiselno podobno se razdaje obeh pogledov upoštevajo tudi za diskretne atribute in druge mere razdaje, ki jih lahko uporablja ReliefF. Ocene W_1 se nato izračuna po enakem postopku kot pri algoritmu ReliefF, le da kot oznake razredov uporabljamo oznake gruč G_1. Enak postopek se ponovi še za W_2, kjer za oznake razredov uporabljamo oznake gruč G_2.

mvRelifeF-mh (Multi View ReliefF - Multi Hit). Posebnost mvReliefF-mh je v definiciji zadetkov in pogreškov. Pri algoritmu Relief zadetek a-tega primera spada v isti razred kot a-ti primer in pogrešek a-tega primera ne spada v razred a-tega primera. Pri mvReliefF-mh je pa zadetek a-tega primera definiran kot primer, ki v obeh pogledih (v obeh gručenjih G_1 in G_2) spada v isto skupino kot $a-ti$ primer. Prav tako je pogrešek a-tega primera tak najbližji primer, ki v nobenem pogledu (v nobenem gručenju G_1 in G_2) ne spada v isto skupino kot a-ti primer. Poostren pogoj za zadetke in pogreške zmanjša množico potencialnih zadetkov in pogreškov. Če si novo definirane zadetke in pogreške razlagamo z zunanjimi merami gručenja iz razdelka 2.3.2, je število možnih zadetkov enako številu RP (resničnih pozitivnih primerov) med G_1 in G_2 ter število možnih pogreškov enako številu RN (resničnih negativnih primerov) med G_1 in G_2. Zato mora mvReliefF-mh za uspešno delovanje izpolnjevati dodatna pogoja:

$$\begin{aligned} RP(G_1, G_2) &\geqslant k, \\ RN(G_1, G_2) &\geqslant k, \end{aligned} \tag{4.2}$$

saj algoritem v vsaki iteraciji poišče k pogreškov in zadetkov. Druga izvedba algoritma tako ocenjuje prispevek atributov le na parih primerov, o katerih se obe gručenji strinjata.

mvReliefF-mdmh (Multi View ReliefF - Multi Distance, Multi Hit) je kombinacija prve in druge izvedbe. Za izračun razdalj med objekti upošteva

značilke iz obeh pogledov, za zadetke in pogreške pa uveljavlja pogoje iz druge izvedbe Multi View ReliefF.

V poglavju 5 preverimo uspešnost vseh treh izvedb na generiranih umetnih podatkih in javno dostopnih podatkovnih zbirkah.

4.3 Ansambelske tehnike

Rezultat predlagane tehnike izbire značilk pri večslojnem gručenju so podmnožice značilk obeh pogledov in najbolj ujemajoči se gručenji. Kljub temu, da se vrnjeni gručenji v splošnem ujemata, je za razlago posameznih gruč ali za primerjavo z drugimi tehnikami gručenja potrebno enotno gručenje. To dosežemo z ansamblom gruč.

Pri nadzorovanem učenju so ansambli tehnike izgradnje modela, ki temelji na rezultatih večih klasifikatorjev ali regresorjev in deluje kot uteženo glasovanje rezultatov. Znane tehnike ansamblov so bagging, boosting, naključni gozd in stacking [12].

Na področju nenadzorovanega učenja pravimo ansambelskim tehnikam tudi konsenzno gručenje. Cilj konsenznega gručenja je na podlagi več gručenj, pridobljenih z različnimi tehnikami gručenja ali z različnimi podatki, ustvariti enotno gručenje, ki je kompromis med rezultati vseh gručenj. Metode konsenznega gručenja rezultat C_e pridobijo z optimizacijo naslednje funkcije [44]:

$$C_e = \underset{\hat{C}}{\operatorname{argmax}} \sum_{i=1}^{k} NMI(\hat{C}, C_i), \tag{4.3}$$

kjer C_1, C_2, C_3 predstavljajo vhodna gručenja in C_e rezultat. Metode konsenznega gručenja skušajo najti novo gručenje C_e, za katerega dobimo maksimalno vsoto ujemanja z ostalimi gručami. Za oceno ujemanja je v članku [44] uporabljena normalizirana medsebojna informacija, ki je opisana v razdelku

2.3.2. V naši metodologiji smo uporabili razčlenjevalno metodo na podlagi podobnosti (angl. Cluster-based similarity partitioning algorithm - CSPA) [41]. Ta za vsako gručenje izračuna binarno matriko podobnosti, kjer vrednost 1 predstavlja par, ki spada v isti gruči, in 0 par, ki spada v različni gruči. Matrike podobnosti se nato sešteje in povpreči s številom gručenj. Nazadnje se novo dobljeno matriko podobnosti uporabi za ponovno gručenje s poljubno tehniko gručenja, ki temelji na podobnosti primerov.

4.4 Večopisne razlage gruč

Zadnja faza predlagane metodologije je večopisna, človeku razumljiva razlaga dobljenih gruč, posebej z značilkami pogleda $S_1 \subseteq P_1$ in posebej z značilkami pogleda $S_2 \subseteq P_2$. Smiselni opisi gruč z večih pogledov omogočajo lažje razumevanje nastanka gruč in povezav med značilkami različnih pogledov.

Razlage gruč temeljijo na izgradnji napovednih modelov na oznakah gruč, ki jih obravnavamo kot oznake razredov. Metodologija uporablja dva načina razlage gruč:

1. metoda RIPPER, ki gruče opiše z odločitvenimi pravili,
2. razlaga prediktorjev (Robnik-Šikonja in Kononenko, 2008), ki kot razlago ponudi prispevke posameznih značilk k odločitvi za neko oznako.

RIPPER (angl. Repeated Incremental Pruning to Produce Error Reduction) [7] je metoda za učenje pravil. Uči se odločitvenih pravil z iterativnim rezanjem za zmanjševanje napake (angl. reduced error pruning). Rezultat algoritma odločitvena pravila, ki elemente razvrščajo v kategorije glede na vrednosti značilk. Primer pravil na podatkovni zbirki IRIS [1] je sledeč:

```
(Petal.Length <= 1.9) => Species=setosa (50.0/0.0)
(Petal.Width <= 1.7) and (Petal.Length <= 4.9) =>
 Species=versicolor (48.0/1.0)
=> Species=virginica  (52.0/3.0)
```

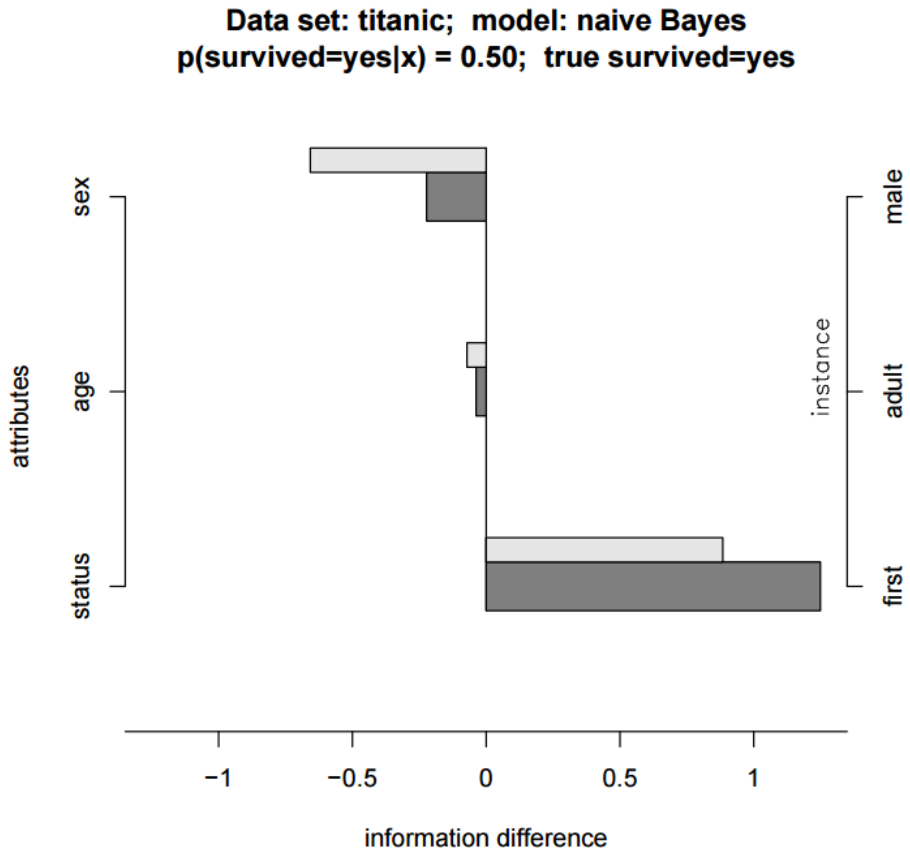
Zadnja vrstica (=>Species=virginica) uvršča primere v skupino virginica v primeru, da nobeden izmed zgornjih pogojev ni bil izpolnjen. Števila v oklepajih označujejo število pravilno oz. napačno klasificiranih primerov določenega razreda. Pravila podajajo človeku razumljivo razlago razdelitve podatkovne zbirke v tri kategorije.

V predlagani metodologiji zgradimo dva niza pravil, enega na podatkih S_1 in drugega za S_2. Kot oznake razredov uporabljamo rezultat ansambla gruč. Takšna pravila predstavljajo človeku razumljivo povezavo med značilkami z različnih pogledov.

Robnik-Šikonja in Kononenko predlagata metodo razlage predikcij, ki ponuja vizualno predstavitev vpliva vrednosti značilk [39]. Metoda prejme učni model, učne podatke in pripadajoče oznake razredov. Na podlagi vpliva spreminjanja vrednosti značilk v modelu skuša sklepati o pomembnosti in vplivu posameznih vrednosti značilk na uvrstitev primerov v razrede. Za vrednost značilke izračuna spremembo informacije (angl. information difference), ki je izražena kot pozitivna ali negativna vrednost vpliva na odločitev za določen razred.

Slika 4.2 prikazuje vpliv značilk *status*, *age*, *sex* na uvrstitev primerov v razred *yes* (preživel). Svetlo sivi stolpični grafi vizualizirajo povprečen vpliv vrednosti značilke na uvrstitev primera v razred *yes* v povprečju za vse primere. Temno sivi stolpični grafi vizualizirajo vpliv vrednosti značilke na uvrstitev konkretnega primera v razred *yes*. Razvidno je, da vrednost $first$ značilke *class* močno prispeva k možnosti preživetja potnika. Vpliv značilk lahko razložimo za posamezne primere ali za celotno podatkovno zbirko.

S predlagano metodologijo za vsako gručo vizualiziramo vpliv značilk na uvrstitev primerov v gruče posebej za S_1 in posebej za S_2. Vizualizacije predstavljajo večopisne razlage gruč in lahko služijo kot človeku razumljive povezave med značilkami z različnih pogledov.



Slika 4.2: Vizualizacija vpliva vrednosti značilk na uvrstitev primera v razred

V naslednjem poglavju testiramo predstavljeno metodologijo na umetnih podatkih, javnih podatkovnih zbirkah iz repozitorija UCI in nazadnje na ADNI podatkovni zbirki pacientov z Alzheimerjevo boleznijo.

Poglavje 5

Empirična evalvacija

Uspešnost predlagane metodologije preverimo z empirično evalvacijo. Sestavljajo jo trije sklopi:

- testiranje na umetnih podatkih,
- testiranje na UCI podatkovni zbirki,
- testiranje na ADNI podatkih pacientov z Alzheimerjevo boleznijo.

Pri testiranju na umetnih podatkih zgradimo umetno učno množico z generiranjem točk iz normalne porazdelitve, za katero vnaprej vemo, kakšne značilke naj bi izbirala uspešna metoda izbire značilk z večimi pogledi. Podatkovno zbirko sestavljata dva pogleda. Vsak pogled vsebuje dve skupini značilk, ki ločujeta primere v dve različni skupini. Prva skupina značilk v vsakem pogledu ločuje primere tako, da se gruče primerov med pogledi strinjajo. Druga skupina značilk v vsakem pogledu ločuje primere v različne skupine in se gruče z uporabo teh skupin značilk ne ujemajo. Pri testiranju na tej podatkovni zbirki ne želimo izbrati le značilk, s katerimi dobimo uspešno gručenje, temveč takšne značilke, ki vplivajo na nastanek takšnih gručenj, ki se med pogledi strinjajo.

Predlagano metodologijo lahko uporabljamo tudi kot samostojno metodo gručenja, saj rezultat ansambla gručenj predstavlja gručenje, ki ga lahko primerjamo z ostalimi tehnikami gručenj. Predlagano metodologijo izvedemo na

podatkovni zbirki Multiple Features Data Set (Handwritten Digits) z UCI ML repozitorija [27].

Področje Alzheimerjeve bolezni zaradi večopisnosti in visokodimenzionalnosti podatkov predstavlja primeren izziv za predlagano metodologijo. Iz podatkovnih zbirk ADNI ustvarimo združeno podatkovno zbirko, na kateri z uporabo predlagane metodologije dobimo večopisno razložene podskupine pacientov. Smiselnost razlag in zaznane povezave med značilkami različnih pogledov ovrednoti strokovnjakinja s področja nevrologije.

5.1 mvReliefF na umetnih podatkih

Razdelek opisuje proces testiranja vseh treh predlaganih izvedb algoritma mvReliefF na umetnih podatkih, za katere vnaprej vemo, katere značilke mora izbrati uspešna metoda izbire značilk z večimi pogledi. Vsak pogled podatkovne zbirke je strukturiran tako, da lahko z njega generiramo več statistično smiselnih gručenj, a le ena podmnožica značilk v vsakem pogledu prispeva k ujemanju gručenj med pogledi. Cilj metod izbire značilk z večimi pogledi je torej v vsakem pogledu poiskati to množico značilk. Možnost večih smiselnih gručenj v vsakem pogledu smo dodali, da preverimo uspešnost izbire značilk z večimi pogledi, saj bi v primeru, kjer je možno le eno smiselno gručenje za vsak pogled, za uspešno izbiro značilk zadostovale klasične metode izbire značilk.

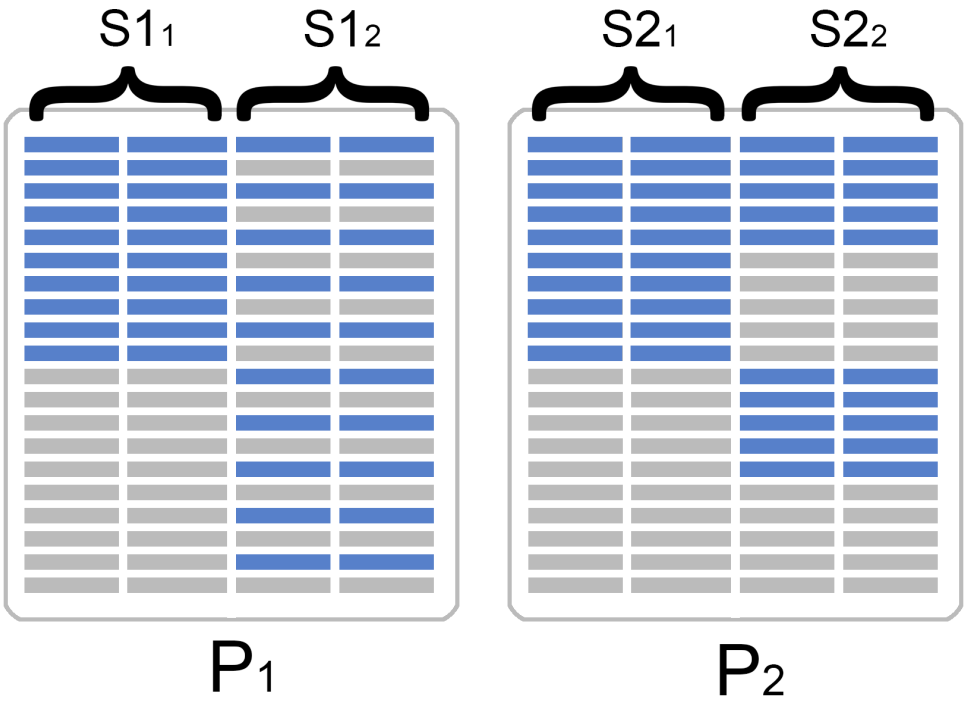
Podatkovna zbirka vsebuje dva pogleda P_1 in P_2. Vsak pogled vsebuje dve skupini značilk, ki primere smiselno ločujeta, a v drugačne razdelitve gruč. Prva skupina značilk v vsakem pogledu ločuje primere tako, da se gručenji med pogledi strinjata. Druga skupina značilk v vsakem pogledu ločuje primere v različne skupine in tako se gručenji z uporabo teh skupin značilk ne strinjata.

Gručenja delijo primere v dve gruči, A in B. Generirali smo 100 primerov.

Značilke smo dobili z generiranjem naključnih vrednosti iz normalne porazdelitve. Primere skupine A imajo vrednosti značilk z jedrom v 1, torej $N(1, \sigma)$. Primeri skupine B imajo značilke s središčem v -1, torej $N(-1, \sigma)$. Prva skupina značilk v prvem pogledu $S1_1$ ima za prvih 50 primerov porazdelitev $N(1, \sigma)$ in za drugih 50 primerov porazdelitev $N(-1, \sigma)$. Enake porazdelitve za prvih 50 in drugih 50 primerov ima prva skupina drugega pogleda $S2_1$. Druga skupina značilk drugega pogleda $S1_2$ ima za sode primere vrednosti porazdeljene $N(1, \sigma)$ in za lihe primere vrednosti $N(-1, \sigma)$. Druga skupina značilk v drugem pogledu ima za prvih pet primerov vrednosti porazdeljene $N(1, \sigma)$ in za naslednjih pet primerov vrednosti porazdeljene $N(-1, \sigma)$ in tako zaporedoma naprej. $S1_1$ in $S2_1$ tako podobno ločujejo primere v dve skupini, $S1_2$ in $S2_2$ pa ne. Vsaka skupina značilk zase smiselno ločuje primere, a le metoda, ki bo iz prvega pogleda izbrala $S1_1$ in iz drugega $S2_1$, bo uspešna v smislu učenja z večimi pogledi.

Vsaka skupina ($S1_1$, $S1_2$, $S2_1$, $S2_2$) vsebuje 40 značilk, ki so bile generirane z normalne porazdelitve z različnimi σ. Vsaka začetna značilka skupine je generirana s $\sigma = 1.1$. σ vsake naslednje značilke se pomnoži z 1.05, kar pomeni, da je σ zadnje (štiridesete) značilke enaka $1.1 * 1.05^{39}$. Zadnje značilke skupin so torej tako razpršene, da je iz njih težko razločiti ali ima porazdelitev središče v 1 ali -1. Šum smo dodali, da pri evalvaciji testiramo tudi sposobnost izbire značilk glede na ločevanje gruč med seboj in ne samo glede na doprinos k ujemanju med gručenji. Shemo porazdelitve prikazuje slika 5.1.

Podatkovna zbirka stotih primerov torej vsebuje 160 značilk, kjer vsak pogled vsebuje polovico značilk. Vsak pogled sestavljata dve skupini značilk, kjer vsako sestavlja 40 značilk. Vsaka skupina je zadostna za gručenje, a le ena skupina vsakega pogleda lahko prispeva k ujemanju gručenj med pogledi. Pričakovan rezultat izbire značilk z večimi pogledi sta skupini značilk $S1_1$ in $S2_1$, ki obe prvo polovico primerov uvrstita v eno skupino in drugo polovico v drugo.



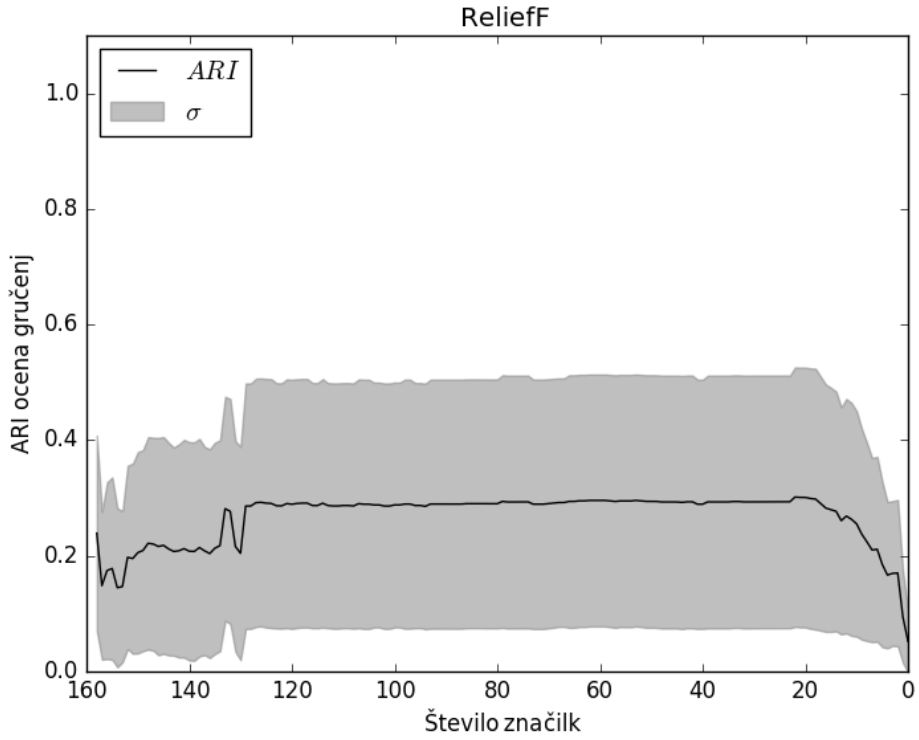
Slika 5.1: Predstavitev podatkov umetne podatkovne zbirke. P_1 in P_2 predstavljata pogleda. Modri pravokotniki predstavljajo vrednosti značilk s porazdelitvijo $N(1, \sigma)$ in sivi pa značilke s porazdelitvijo $N(-1, \sigma)$. $S1_{1,2}$ in $S2_{1,2}$ predstavljajo skupine značilk, ki zadostujejo za smiselno gručenje.

Naslednji podrazdelki od 5.1.1 do 5.1.4 vsebujejo rezultate izvajanja predlagane metodologije z uporabo vseh treh izvedb mvReliefF in klasičnega ReliefF. Uporabili smo metodo spektralnega gručeja. Test vsake izvedbe algoritma smo izvedli desetkrat in rezultate povprečili in poleg ocene ujemanj gručenj izrisali tudi standardno deviacijo.

5.1.1 ReliefF

Najprej smo za izbiro značilk uporabili algoritem ReliefF, da bi potrdili, da podatkovna zbirka predstavlja problem izključno za izbiro značilk z večimi pogledi. Ker ima vsak pogled več možnih smiselnih gručenj, je možnost, da

bi s klasičnimi metodami izbrali značilke, ki prispevajo k skupnemu gručenju, majhna.



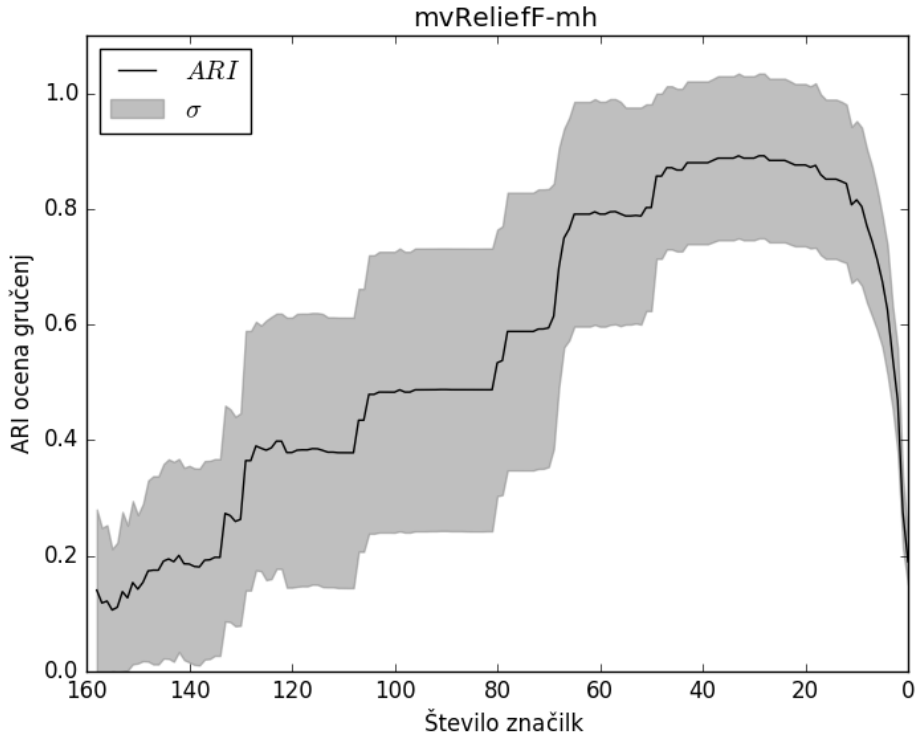
Slika 5.2: Proces izbire značilk pri večslojnem gručenju na umetnih podatkih z uporabo algoritma ReliefF. *ARI* (med 0 in 1, večja vrednost pomeni boljše ujemanje med gručenji) predstavlja povprečno vrednost ujemanja gručenj med pogledi v posamezni fazi izbire značilk. σ predstavlja standardni odklon ujemanja gručenj vseh desetih poskusov izvajanja testa.

Slika 5.2 prikazuje stopnjo ujemanja gručenj med pogledi (izraženo z ARI oceno) v odvisnosti od števila značilk pri procesu izbire značilk. Na primer, na območju med 130 in 20 značilk je bila povprečna ARI ocena ujemanja gručenj približno 0.3. ReliefF je v vseh desetih poskusih uspešno izbiral značilke, ki ustvarjajo dobro ločene gruče, a v le dveh poskusih je izbral takšne značilke, da sta se gručenji med pogledi ujemali. S slike je razvidno,

da proces izbire značilk z uporabo ReliefF v povprečju občutno ni izboljšal ujemanja med gručenji. To pomeni, da podatkovna zbirka predstavlja primeren problem za metodo izbire značilk z več pogledi.

5.1.2 mvReliefF-mh

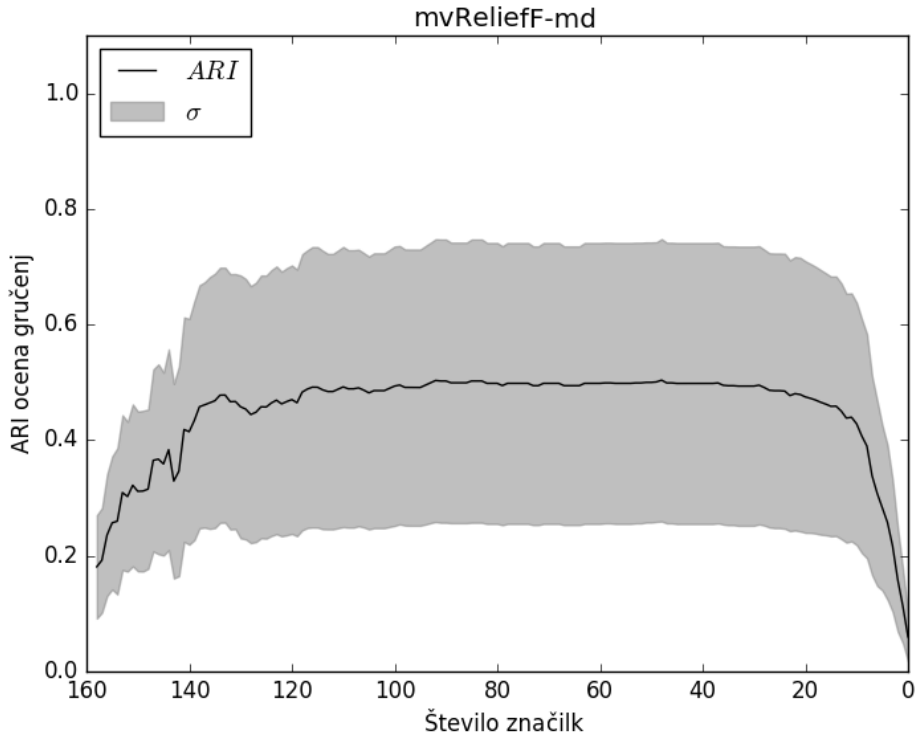
Algoritem mvReliefF-mh (Multi View ReliefF - Multi Hit, glej str. 45) je izvedba algoritma ReliefF, ki za zadetke upošteva le pare primerov, ki spadajo v iste gruče v obeh pogledih.



Slika 5.3: Proces izbire značilk pri večslojnem gručenju na umetnih podatkih z uporabo mvReliefF-mh. *ARI* predstavlja povprečno vrednost ocene ujemanja gručenj med pogledi v posamezni fazi izbire značilk. σ predstavlja standardni odklon ujemanja gručenj vseh desetih poskusov izvajanja testa.

S slike 5.3 je razvidno, da je mvReliefF-mh v skoraj vsakem izmed desetih testov uspešno izbral pričakovano skupino značilk, s katero dobimo uspešno ujemanje gručenj. Najboljša povprečna ARI ocena je 0.89. Rezultat ujemanja gruč pri majhnem številu značilk se približuje ničli, saj smo značilke generirali iz normalne porazdelitve s precej veliko varianco (najboljša značilka vsake skupine ima porazdelitev $N(\mu, 1.1)$). Tako je za uspešno gručenje potrebno vsaj nekaj značilk iz vsake skupine.

5.1.3 mvReliefF-md

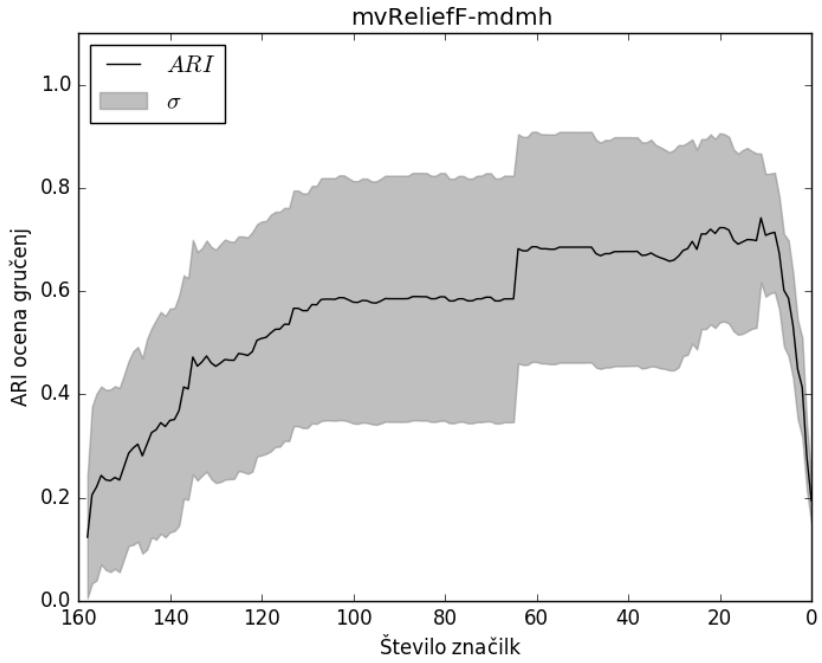


Slika 5.4: Proces izbire značilk pri večslojnem gručenju na umetnih podatkih z uporabo mvReliefF-md. ARI predstavlja povprečno vrednost ocene ujemanja gručenj med pogledi v posamezni fazi izbire značilk. σ predstavlja standardni odklon ujemanja gručenj vseh desetih poskusov izvajanja testa.

Algoritem mvReliefF-md (Multi View ReliefF - Multi distance, glej str. 44) je izvedba algoritma ReliefF, ki pri oceni značilk posameznega pogleda za računanje razdalj med objekti upošteva značilke iz obeh pogledov.

Pri desetkratnem testiranju na umetnih podatkih je dosegel najboljšo povprečno *ARI* oceno 0.50, kar je najslabše ujemanje med vsemi tremi različicami mcReliefFa. Optimalno ujemanje gručenj je ta inačica dosegla le pri petih poskusih izvajanja testa, pri ostalih izvajanjih testa pa k ujemanju ni prispevala.

5.1.4 mvReliefF-mdmh



Slika 5.5: Proces izbire značilk pri večslojnem gručenju na umetnih podatkih z uporabo mvReliefF-mdmh. *ARI* predstavlja povprečno vrednost ocene ujemanja gručenj med pogledi v posamezni fazi izbire značilk. σ predstavlja standardni odklon ujemanja gručenj vseh desetih poskusov izvajanja testa.

Algoritem mvReliefF-mdmh (Multi View ReliefF - Multi Distance Multi Hit, glej str. 45) je izvedba algoritma ReliefF, ki upošteva razdalje iz obeh pogledov in ujemanje zadetkov in pogreškov.

Pri desetkratnem testiranju je dosegel najboljšo povprečno ARI oceno 0.74, kar je bolje kot mvReliefF-md, a slabše kot mvReliefF-mh.

Pri testiranju treh izvedb mvReliefF smo spoznali, da vse tri izvedbe uspešno najdejo značilke, ki prispevajo k ujemanju med gručenji. Najbolje se je izkazala metoda mvRelief-mh.

S testiranjem algoritmov na umetnih podatkih smo preverili uspešnost le za prvo fazo predlagane metodologije, ki je na sliki 4.1 prikazana kot zgornji pravokotnik. Naslednji razdelek preveri izvajanje algoritma vključno z ansambli gručenj in primerja rezultate z obstoječo tehniko gručenja z večimi pogledi [24].

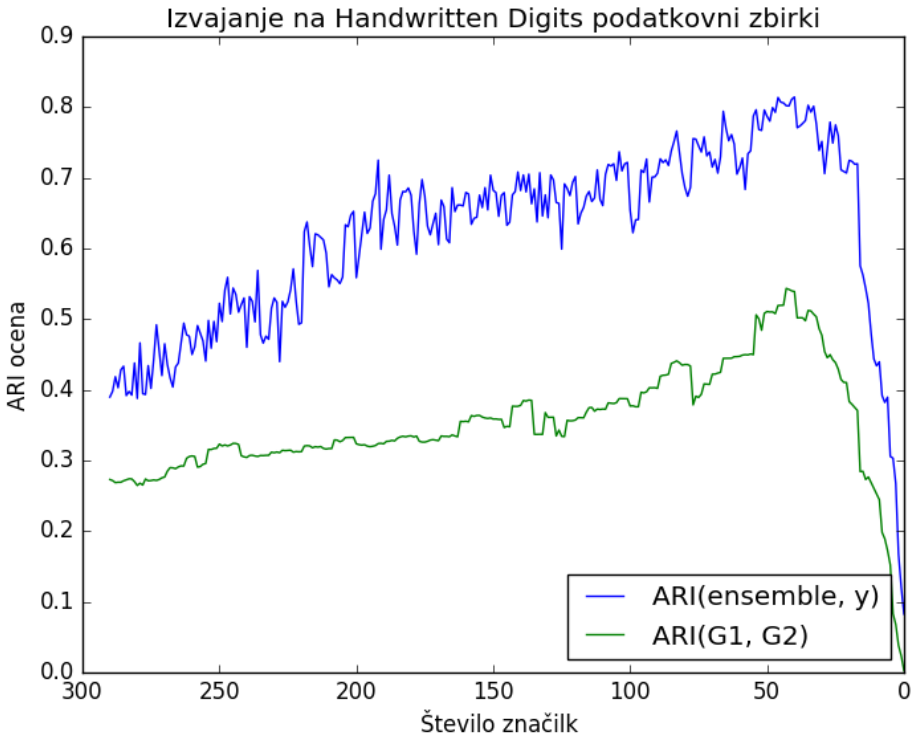
5.2 Predlagana metodologija kot metoda gručenja

Predlagana metodologija skuša za pogleda P_1 in P_2 najti podmnožici značilk, s katerima dobimo gručenji, ki se med seboj najbolj strinjata. Če sta si gručenji dovolj podobni, lahko iz njiju z uporabo ansambelskih tehnik dobimo enotno gručenje, na kar lahko gledamo kot na novo metodo gručenja z večimi pogledi, saj za vhodne podatke P_1, P_2 in algoritem gručenja C vrne oznake gruč G.

V naslednjem razdelku opišemo rezultate kratkega poskusa na manjši podatkovni zbirki opisani z več pogledov in rezultate primerjamo z rezultati objavljenimi v članku [24].

Kumar in Daumé (2016) [24] predlagata novo metodo gručenja z več pogledi, ki temelji na spektralnem gručenju. Uspešnost metode se testira na treh znanih podatkovnih zbirkah za učenje z več pogledi Handwritten

digits, BBC data in Reuters multilingual data, ki so preneseni s spletnega repozitorija UCI [27]. Handwritten digits je edina podatkovna zbirka, ki ne zahteva preprocesiranja, postopek izvajanja algoritma je pa podrobno dokumentiran in rezultati so objavljeni v članku. Tako smo tudi mi uspešnost predlagane metode preizkusili na podatkovni zbirki Handwritten digits.



Slika 5.6: Povprečni rezultati desetkratnega izvajanja predlagane metodologije na podatkovni zbirki Handwritten Digits. Modra črta $ARI(G1, G2)$ predstavlja stopnjo ujemanja gručenj med pogledi. $ARI(ensemble, y)$ predstavlja stopnjo ujemanja ansambla, ki smo ga dobili z združevanjem gručenj z obeh pogledov, in dejanskih oznak razredov y.

Podatkovna zbirka vsebuje 2000 primerov na roko napisanih številk, ki so opisane s 76 značilkami v enemu pogledu in 216 značilkami v drugem pogledu. Za primere poznamo dejanske oznake razredov y, kar pomeni, da lahko z zu-

nanjimi kriteriji ocenjujemo uspešnost gručenja glede na pričakovane oznake njih.

Kot gručenje smo uporabili metodo spektralno gručenje [31] (glej str. 16) in za izbiro značilk algoritem mvReliefF-mh. Pri vsaki iteraciji izbire značilk smo izračunali ujemanje gručenj med pogledi in ujemanje ansambla in dejanskih oznak y. Test smo izvedli desetkrat.

S slike 5.6 je razvidno, da stopnja ujemanja gruč med pogledi močno korelira s stopnjo ujemanja ansambla gruč in pričakovanih oznak primerov. Najboljšo ARI oceno 0.814 smo dobili s približno 40 spremenljivkami.

F-score je mera, ki se tipično uporablja pri dvorazrednih klasifikacijskih problemih. Pfitzner in sod. (2009) predlagajo razširitev mere za gručenje [34]. Ta za dve gručenji $G1$ in $G2$ izračuna spremenljivke $a = RP$, $b = LN$ in $c = LP$, kjer RP, LN in LP predstavljajo vrednosti razložene v 2.3.2. Nato F-score izračunamo s formulo $F = \frac{2a}{2a+b+c}$. Mero smo implementirali z namenom celovite primerjave naših rezultatov z rezultati objavljenimi v članku [24]. S tabele 5.1 je razvidno, da so rezultati našega gručenja boljši od rezultatov iz članka, saj uporabljene mere kažejo na boljše ujemanje dobljenih gruč z dejanskimi oznakami.

Tabela 5.1: Rezultati ujemanja gručenj z dejanskimi oznakami y za podatkovno zbirko Handwritten digits. Ocene so popravljen Randov indeks - ARI, normalizirana medsebojna informacija - NMI in F-score. Vrstica Co-trained spectral predstavlja rezultate iz članka [24], vrstica Spectral mvReliefF-mh pa predstavlja rezultate z uporabo spektralnega gručenja in mvReliefF-mh. Rezultati v oklepajih predstavljajo standardno deviacijo rezultatov izvajanj.

Metoda	**ARI**	**NMI**	**F-score**
Spectral (mvReliefF-mh)	0.814 (0.036)	0.842 (0.030)	0.833 (0.032)
Co-trained spectral	0.695(0.054)	0.765 (0.031)	0.726 (0.048)

Rezultati kažejo, da predlagana metodologija z optimizacijo stopnje uje-

manja gručenj med pogledi izboljša tudi ujemanje ansambla gručenj s pričakovanimi oznakami y in posledično izboljša rezultat gručenja.

5.3 Evalvacija na ADNI podatkih

Podatki v medicini so pogosto zajeti na več načinov, z večih pogledov. Na področju Alzheimerjeve bolezni, ki predstavlja kritično, še neozdravljivo obliko demence, imamo o pacientih podatke zajete na veliko načinov, ki jih lahko v grobem uvrstimo na biološke in klinične preiskave. Vpliv in pomen določenih značilk na stanje bolezni nam je precej znan, a razlaga povezav med kliničnimi in biološkimi značilkami skupin pacientov z Alzheimerjevo boleznijo je manj raziskano področje. Ker ADNI podatkovna zbirka vsebuje podatke več sto različnih vrst preiskav pacientov z Alzheimerjevo boleznijo, predstavlja primeren izziv za izvajanje predlagane metodologije.

Namen naše evalvacije je najti smiselne podskupine pacientov z Alzheimerjevo boleznijo, ki jih lahko opišemo na več načinov, s kliničnimi in biološkimi značilkami posebej. Večopisna narava rezultatov strokovnjakom omogoča razumevanja nastanka gruč in povezav med značilkami ter pogledi.

5.3.1 Priprava podatkovne zbirke

Podatkovno zbirko ADNI sestavlja več sto ločenih preiskav, ki vključujejo podatke rentgenskih slik, genetsko analizo, klinične teste, fiziološke lastnosti pacientov, krvne analize in mnogo drugih preiskav. V grobem lahko značilke razdelimo na klinične in biološke. Ker je povprečen pacient opravil le nekaj omenjenih raziskav, bi bila združena podatkovna zbirka, ki bi vsebovala vse možne značilke, zelo redko zapolnjena (vsebovala bi veliko večino manjkajočih vrednosti).

Priprava podatkovne zbirke temelji na združevanju posameznih preiskav. V izogib redko zapolnjeni podatkovni zbirki smo izbrali takšne preiskave, za katere pri čim več pacientih velja, da so opravili vse izmed izbranih preiskav.

Tabela 5.2: Raziskave uporabljene za pripravo podatkovne zbirke

Preiskava	**Št. značilk**	**Kategorija**	**Opis**
ucberkeleyfdg	31	biološka	podatki FDG pozitronske emisijske tomografije
adnimerge	11	biološka	tabela, ki združuje glavne ADNI preiskave
medhist	19	klinična	podatki o preteklih boleznih pacienta
cdr	7	klinična	klinični test demence
moca	36	klinična	klinični test kognitivnih sposobnosti
uwnpsychsum	2	klinična	nevropsihološki klinični test
adas	15	klinična	klinični test Alzheimerjeve bolezni
mmse	32	klinična	klinični test kognitivnih sposobnosti
adnimerge	27	klinična	tabela, ki združuje glavne ADNI preiskave

Tabele preiskav so bile prenesene z ADNI [30] repozitorija. Stolpci predstavljajo značilke in vrstice posamezne obiske pacientov. Vsak obisk vsake preiskave je unikatno definiran s primarnim RID in sekundarnim $VISCODE$ ključem. RID predstavlja unikatni identifikator pacienta in $VISCODE$ kodo obiska pacienta, saj je pacient lahko na preiskave hodil v različnih fazah bolezni. Tabele posameznih preiskav smo združevali po principu notranjega združevanja (angl. inner join). To pomeni, da smo pri združevanju obeh tabel združili in obdržali le vrstice, kjer sta se primarni in sekundarni ključ obeh tabel skladala. Tako smo obdržali le paciente, ki so opravili preiskave iz obeh tabel, ki jih združujemo. S tem smo preprečili nastanek dodatnih manjkajočih vrednosti, a obenem z vsakim združevanjem potencialno zmanjšali število primerov v zbirki. Značilke vsake preiskave smo uvrstili med biološke

ali klinične značilke. Tabela 5.2 vsebuje preiskave, uvrstitev in število značilk končne podatkovne zbirke.

Tabela adnimerge je posebnost, saj že sama tabela združuje številne ADNI preiskave in vključuje tako biološke kot klinične značilke, zato smo jo razbili na dva dela. Pri preiskavi ucberkeleyfdg smo podatke preprocesirali. Obiski pacientov so predstavljeni v več vrsticah, kjer vsaka vrstica predstavlja analizirano področje možganov. Ustvarili smo nove značilke, kjer se vsaka značilka nanaša na določeno področje možganov in tako podvojene vrstice obiskov pacientov združili v eno.

Vsak obisk pacienta vsebuje tudi diagnozo DX, ki je natančneje opisana v razdelku 3.2. Iz združene podatkovne množice smo odstranili obiske pacientov z diagnozo *normalno stanje* in s tem pridobili podatkovno zbirko z izključno bolnimi pacienti.
Ker je nekaj značilk vsebovalo precejšen delež manjkajočih vrednosti, smo značilke, ki so imele več kot 50% manjkajočih vrednosti odstranili. Končna podatkovna zbirka vsebuje podatke 136 pacientov. Vsebuje 170 značilk; 129 kliničnih, ki predstavljajo pogled P_1 in 41 bioloških, ki predstavljajo pogled P_2. V zbirki je 602 manjkajočih vrednosti.

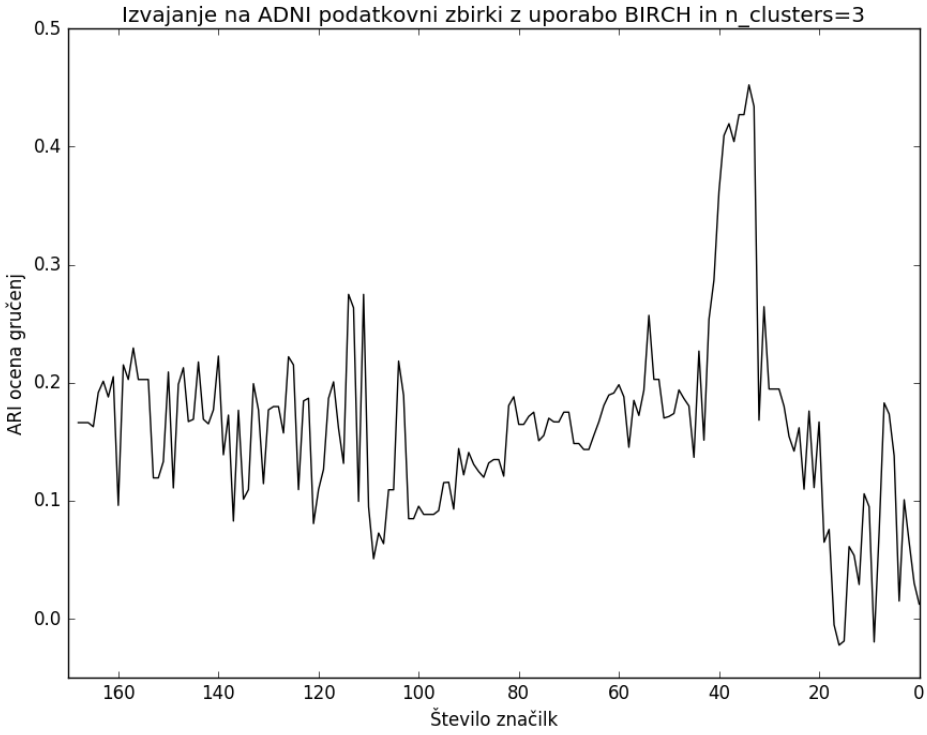
5.3.2 Izvajanje metodologije

Najprej smo podatkovno zbirko pripravili z zamenjavo manjkajočih vrednosti. Uporabili smo zamenjavo s povprečenjem, kar pomeni, da smo manjkajoče vrednosti zamenjali s povprečnimi vrednostmi značilk.

Predlagana metodologija večličnega gručenja z razlago kot vhodne parametre prejme pogleda P_1 in P_2 ter algoritem gručenja $C()$. S poskušanjem različnih metod gručenja, smo najboljše rezultate dobili z uporabo algoritma BIRCH [49] (glej str. 16) in številom gruč 3. Za izbiro značilk smo uporabili mvReliefF-mh.

Slika 5.7 prikazuje postopek izbire značilk z uporabo algoritma BIRCH pri številu gruč 3. Graf prikazuje ARI oceno ujemanja gručenj med pogledi

glede za dano število značilk. Najboljšo ARI oceno **0.452** ujemanja gručenj smo dobili pri 17 kliničnih in 19 bioloških značilkah. Zaradi relativno majhnega števila primerov (136) je stabilnost metod gručenja majhna. To vpliva na stopnjo ujemanja gručenj, zato je z grafa razvidno močno nihanje ocene ARI. Vseeno pa lahko opazimo pozitiven trend ARI ocene na intervalu od 100 do 36 značilk.



Slika 5.7: Izbira značilk na združeni ADNI podatkovni zbirki z uporabo BIRCH in številom gruč 3. Za izbiro značilk smo uporabili mReliefF-mh. ARI predstavlja oceno ujemanja gručenja na bioloških podatkih in gručenja na kliničnih podatkih.

Izbrane biološke značilke: CDRSB, MMSE, CDJUDGE, CDGLOBAL, MMSTATE, MMTRIALS, MMWATCH, MMWRITE, MMSCORE,

Q2SCORE, Q6SCORE, Q9SCORE, Q10SCORE, Q11SCORE, Q12SCORE, TOTSCORE, TOTAL13.

Izbrane biološke značilke: FDG, AngularLeftMAX, AngularLeftMEAN, AngularLeftMEDIAN, AngularLeftMIN, AngularRightMAX, AngularRightMEAN, AngularRightMEDIAN, AngularRightMIN, CingulumPostBilateralMAX, CingulumPostBilateralMEAN, CingulumPostBilateralMEDIAN, TemporalLeftMAX, TemporalLeftMEAN, TemporalLeftMEDIAN, TemporalLeftMIN, TemporalLeftMODE, TemporalRightMAX, TemporalRightMEDIAN.

Pomen posameznih značilk je opisan v prilogi C.

V naslednji fazi smo s pomočjo ansambla dobili enotno gručenje na podlagi gručenj, ki sta dosegli najboljšo oceno ARI 0.452. Uporabili smo metodo CSPA, ki je opisana v razdelku 4.3. Deluje na principu enačbe (4.3), ki skuša maksimiririrati vsoto ujemanja posameznih gruč in novo nastale gruče - ansambla. ARI ocena med gručenjem na bioloških značilkah in dobljenim ansamblom je 0.92, ARI ocena med gručenjem na kliničnih značilkah in ansamblom je pa 0.52. To pomeni, da je ansambel uspešen, saj je podoben obema gručenjema.

Ansambel gručenj vseh 136 pacientov deli v tri skupine A, B in C.

Skupina A vsebuje 42 pacientov. Izmed teh ima 29 diagnozo *demenca*, 10 diagnozo BKM *do demenca* in 3 pacienti diagnozo BKM. Iz porazdelitve diagnoz pacientov v gruči lahko sklepamo, da gruča večinoma vsebuje paciente s težko obliko bolezni.

Skupina B vsebuje 9 pacientov. Izmed teh ima 5 diagnozo BCM, le eden pa diagnozo *demenca*. Ta skupina torej večinoma vsebuje paciente z blago kognitivno motnjo.

Skupina C vsebuje 85 pacientov. 60 jih ima diagnozo BCM, 11 diagnozo *demenca*, 7 diagnozo *normalno stanje do* BCM, 6 diagnozo BCM *do demenca* in eden diagnozo *normalno do* BCM. Skupina večinoma vse-

buje paciente z blago kognitivno motnjo, a vsebuje tudi nekaj pacientov z demenco in nekaj z blago obliko bolezni.

Naslednji podrazdelek opisuje postopek večopisne razlage dobljenih gruč.

5.3.3 Razlaga gruč pacientov

Za lažje razumevanje nastanka gruč in povezav značilk med pogledi skušamo dobljene gruče razložiti posebej s kliničnimi značilkami in posebej z biološkimi značilkami z uporabo metod razlage prediktorjev. Uporabili smo odločitvena pravila, pridobljena z algoritmom RIPPER [7] in razlago prediktorjev s pozitivnim oz. negativnim vplivom značilk (Robnik-Šikonja in Kononenko, 2008). Skušali smo razložiti gruče ansambla A, B in C.

Pravila algoritma RIPPER so predstavljena v tabeli 5.3:

Tabela 5.3: Odločitvena pravila algoritma RIPPER za razlago gruč pacientov posebej z biološkimi in posebej s kliničnimi značilkami.

Gruča	**Biološka razlaga**	**Klinična razlaga**	**Št. p.**
A	(FDG <= 5.39459) (90%)	(TOTSCORE >= 25) (80%)	42
B	(FDG >= 6.98926) (100%)	(CDRSB <= 1) **IN** (TOTSCORE <= 5)(100%) **ALI** (TOTSCORE <= 11) **IN** (TOTSCORE>= 10) (100%)	9
C	(FDG >5.39459) **IN** (FDG <6.98926) (99%)	Noben zgornji pogoj ni izpolnjen	85

Podatki v oklepajih predstavljajo uspešnost odločitvenega modela pri opisu posamezne gruče s pravilom. Pravila se preverjajo zaporedno. To

pomeni, da pri preverjanju pravila za skupino B že predpostavljamo, da pogoj za skupino A ni izpolnjen. V skupino C spadajo primeri, pri katerih noben izmed zgornjih (A in B) pogojev ni bil izpolnjen. Pravilo biološke razlage skupine C je ekvivalentno negaciji pravil A in B.

Gruče smo razložili tudi z metodo razlage prediktorjev (Robnik-Šikonja in Kononenko, 2008). Ta nudi vizualni izris vpliva posameznih značilk na uvrstitev primerov v določeno skupino. Razlage lahko generiramo za posamezne primere ali za celoten model. V prilogah najdemo vizualizacije vpliva značilk, ki vsako gručo opišejo posebej z biološkimi značilkami in posebej s kliničnimi značilkami. V prilogi A najdemo razlage celotnega modela za opis gruč in v prilogi B najdemo razlage za posamezne reprezentativne primere.

Iz razlage prediktorjev je razvidno, da izbrane značilke dobro ločujejo dobljene gruče, tako so podatki statistično smiselni. Ocena uporabnosti in praktične smiselnosti rezultatov je opisana v naslednjem podrazdelku, ki temelji na oceni strokovnjakinje s področja nevrologije.

5.3.4 Komentarji strokovnjakinje s področja nevrologije

Rezultate je ocenila strokovnjakinja s področja nevrologije, ki meni, da so večopisne definicije gruč ter povezave med (sicer že znanimi) kliničnimi in biološkimi značilkami smiselne. V nadaljevanju navajamo nekaj ugotovitev.

Skupina A je izražena z visokimi vrednosti $TOTSCORE$ in $FDG < 5.39$ in smiselno opisuje demenco Alzheimerjevega tipa.

Skupina C je izražena s pravili $5,39 \leq FDG \leq 6.99$ in ($TOTSCORE < 5$ ali $10 \leq TOTSCORE \leq 11$) in smiselno opisuje paciente tipa BKM (angl mild cognitive disorder - MCI).

Zanimiva je skupina B, ki jo opisujeta značilki $FDG < 6,99$ in $11 < TOTSCORE < 25$, saj vsebuje paciente s t.i. predklinično oz. presimptomatsko demenco Alzheimerjevega tipa, kar bi bilo smiselno preveriti še z

dodatno analizo in z vključitvijo več bioloških značilk.

Naše razlage bi bile praktično bolj uporabne, če bi vključevale več različnih vrst bioloških značilk, saj je naša metoda dosegla najboljše ujemanje gručenj s samo eno vrsto bioloških značilk. Ta ugotovitev pomeni, da bi bilo našo metodologijo smiselno vključiti v širši sistem za odkrivanje znanja, ki bi podpiral interakcijo z eksperti in bi preference, ki izhajajo iz ujemanja pogledov dopolnjeval z ekspertnim znanjem strokovnjakov.

Nazadnje je izpostavila, da bi bilo tovrstno metodologijo smiselno preizkusiti na podatkovni zbirki, ki vključuje tudi zdrave paciente, saj naj bi uspešna metoda v prvi fazi znala ločevati zdrave in bolne paciente. Predpostavlja, da bi bila skupina zdravih pacientov opisana s pravili $FDG \geq 6.9$, $TOTSCORE < 9$ in $MMSCORE > 24$. Takšno testiranje metodologije smo izvedli v naslednjem podrazdelku na podatkovni zbirki z bolnimi in zdravimi pacienti.

5.3.5 Ločitev zdravih in bolnih pacientov

Za povečanje zaupanja v predlagano metodologijo smo v ADNI podatkovno zbirko vključili tudi zdrave paciente, s predpostavko, da bo uspešna metoda znala prepoznati skupino zdravih pacientov.

Ta podatkovna zbirka je vsebovala 221 (85 zdravih in 136 bolnih) pacientov in enake značilke kot podatkovna zbirka opisana v podrazdelku 5.3.1.

Da bi proces poenostavili, smo uporabili število gruč 2 in dobili gruči, ki ju označimo z A in B. Uporabljali smo isto metodologijo z istimi algoritmi, kot v razdelku 5.3.2. Pri 37 bioloških in 48 kliničnih značilkah smo dosegli najboljšo ARI oceno 0.499.

Skupina A vsebuje 50 pacientov. Izmed teh ima 34 diagnozo *demenca*, 11 diagnozo *BKM do demenca* in 5 pacientov diagnozo *BKM*. Iz porazdelitve diagnoz pacientov v gruči lahko sklepamo, da gruča vsebuje bolne paciente.

Skupina B vsebuje 171 oseb. Izmed teh jih ima 85 diagnozo $Normalno$, 63 diagnozo BKM, 8 diagnozo $Normalno\ do\ BKM$, ostalih nekaj pacientov pa spada v druge skupine. Iz porazdelitve diagnoz pacientov v gruči lahko sklepamo, da gruča vsebuje zdrave in manj bolne paciente, saj so v gruči prav vsi zdravi pacienti iz podatkovne zbirke. Porazdelitev tudi nakazuje, da so blago bolni pacienti z izbranimi značilkami težko ločljivi od zdravih oseb.

Dobljene gruče smo opisali z uporabo odločitvenih pravil. Kljub temu, da smo optimalno ujemanje gručenj dosegli pri 37 bioloških in 48 kliničnih značilkah, smo dobljene gruče opisali le z značilkami FDG, $MMSCORE$ in $TOTSCORE$ v namen potrditve delovanja metodologije. Tabela 5.4 vsebuje odločitvena pravila dobljenih gruč.

Tabela 5.4: Odločitvena pravila algoritma RIPPER za razlago gruč pacientov posebej z biološkimi in posebej s kliničnimi značilkami.

Gruča	**Biološka razlaga**	**Klinična razlaga**	**Št. p.**
A	(FDG <= 5.373)	(TOTSCORE >= 14) **IN** (MMSCORE <= 24)	50
B	(FDG >5.373)	(TOTSCORE <14) **ALI** (MMSCORE >24)	171

S tabele 5.4 je razvidno, da rezultati odločitvenih pravil predpostavko strokovnjakinje, da bo uspešna metoda našla skupino zdravih pacientov s $FDG \geq 6.9$, $TOTSCORE < 9$ in $MMSCORE > 24$, delno potrjujejo. Za skupino B zdravih (in manj bolnih) pacientov se vrednost pravila za MMSCORE popolnoma ujema s predpostavko. Vrednosti značilk FDG in TOTSCORE od predpostavke odstopajo, a razmerja vrednosti (večje ali manjše) so prava. Rahlo odstopanje vrednosti gre verjetno pripisati dejstvu, da skupina B vsebuje tudi rahlo bolne paciente.

Poskus potrjuje pravilno delovanje metodologije in vodi do zaključka, da bi bilo metodologijo smiselno uporabiti v obširnejši raziskavi, kjer bi bilo ekspertno znanje vključeno že v fazi načrtovanja

Poglavje 6

Sklepne ugotovitve

V nalogi predstavimo novo metodologijo večličnega gručenja z razlago. Vhodna parametra sta podatkovna zbirka, ki je opisana z enim ali več pogledi, in algoritem gručenja. V prvi fazi metodologije skušamo izbrati takšne značilke, ki prispevajo k čim boljšemu ujemanju gručenj med pogledi. To dosežemo z uporabo metode mvReliefF, ki favorizira značilke, ki omogočajo uspešno večlično gručenje. V drugi fazi s tehniko ansamblov združimo dobljena gručenja med pogledi in v tretji fazi razložimo dobljene gruče na človeku razumljiv način posebej z značilkami vsakega pogleda.

Pri empirični evalvaciji metodologije sprva preizkusimo uspešnost na umetnih podatkih. Generiramo jih tako, da so težavni za učenje z večimi pogledi, in pokažemo, da metode izbire značilk z enim pogledom problema ne rešujejo uspešno. Predlagana algoritma mvReliefF-md in mvReliefF-mdmh se izkažeta za uspešni tehniki izbire značilk z večimi pogledi. V naslednji fazi preverimo uspešnost predlagane metodologije kot tehnike gručenja. Na podatkovni zbirki Handwritten digits iz repozitorija UCI izvedemo predlagano metodologijo in naše rezultate ujemanja ansambla gručenj z dejanskimi vrednostmi primerjamo s sorodnim člankom. Rezultati pokažejo, da izbira značilk z večimi pogledi občutno izboljša uspešnost gručenja, saj so dobljeni rezultati, glede na uporabljene mere, boljši kot rezultati v primerjanem članku. V tretjem koraku empirične evalvacije izvedemo metodologijo

na ADNI podatkih pacientov z Alzheimerjevo boleznijo. Ker ADNI vsebuje ločene podatke več raziskav pacientov z Alzheimerjevo boleznijo, smo pripravili združeno podatkovno bazo bioloških (prvi pogled) in kliničnih (drugi pogled) značilk, ki vsebuje čim več preiskav s čim manj manjkajočimi vrednostmi. Na združeni zbirki smo s filtrirno-ovojno metodo izbire značilk našli značilke, s katerimi dobimo dobro ujemajoči se gručenji med pogledi z ARI oceno ujemanja 0.452. Dobljeni gručenji združimo z ansambelsko metodo CSPA in dobljene gruče razložimo z uporabo odločitvenih pravil in z metodo razlage prediktorjev posebej s kliničnimi značilkami in posebej z biološkimi značilkami. Večopisne razlage gruč služijo kot človeku razumljiva razlaga gruč in omogočajo lažje razumevanje povezav med značilkami v različnih pogledih. Dobljene razlage prediktorjev smo posredovali strokovnjakinji na področju nevrologije, ki je rezultate ocenila kot smiselne.

Kljub temu, da se je predlagana metodologija izkazala za uspešno, predlagamo več izboljšav. Predlagana filtrirno-ovojna metoda izbire značilk v vsaki fazi izvede gručenji posebej za vsak pogled, izračuna ujemanja med gručenji ter z uporabo algoritma mvReliefF oceni značilke. Najslabšo značilko odstranimo in proces ponavljamo do ustavitvenega pogoja, ko sta v vsakem pogledu le še dve značilki. Nazadnje vrnemo podmnožici značilk, pri katerih smo dobili najboljšo oceno ujemanja gručenj med pogledi. Tovrsten princip je sicer učinkovit, a časovno potraten, saj se (zaradi odstranjevanja le ene značilke v vsaki iteraciji) iste značilke pri skoraj enakih razdelitvah gruč ocenjujejo večkrat. V nadaljnjih raziskavah bi bilo smiselno metodologijo nadgraditi z uporabo optimizacijskih metod, ki bi same izbrale korak (število odstranjenih značilk v vsaki iteraciji) in tako izboljšale hitrost izvajanja.

Razlog, zakaj za ustavitveni pogoj pri filtrirno-ovojni metodi nismo uporabili strožjega pogoja (na primer, ko se ARI ocena začne manjšati), gre pripisati dejstvu, da je ARI ocena v procesu izbire značilk kljub vidnim trendom naraščanja/padanja precej nemonotona. Tovrsten ustavitveni pogoj bi algoritem verjetno pustil v lokalnem optimumu. V nadaljnjih raziskavah bi

bilo smiselno najti ustreznejši ustavitveni pogoj, ki ne preišče celotnega prostora, a se vseeno izogiba lokalnim optimumom.

Slabost rezultatov na podatkih ADNI je, da so skupine opisane s samo eno vrsto bioloških značilk, saj je metoda izbire značilk dosegla najboljše ujemanje gručenj le pri značilkah raziskave FDG-PET. V nadaljnjem delu bi bilo v metodologijo smiselno vključiti možnost lastne (vsiljene) izbire značilk, s katerimi bi želeli opisati gruče. S tovrstno metodologijo bi lahko izdelali interaktivno orodje za večopisno razlago gruč podatkov.

Literatura

[1] E. Anderson, "The species problem in iris," *Annals of the Missouri Botanical Garden*, vol. 23, no. 3, pp. 457–509, 1936.

[2] P. Berkhin, "A survey of clustering data mining techniques," in *Grouping Multidimensional Data*. Springer, 2006, pp. 25–71.

[3] A. Blum and T. Mitchell, "Combining labeled and unlabeled data with co-training," in *Proceedings of the eleventh annual conference on Computational learning theory*. ACM, 1998, pp. 92–100.

[4] M. Breskvar, "Relating biological and clinical features of alzheimers patients with predictive clustering trees," in *Conference on Data Mining and Data Warehouses*. SiKDD, 2015, pp. 1–4.

[5] A. Burns and I. Steve, "Alzheimer's disease." British Medical Journal, 2009, pp. 1–9.

[6] G. Chandrashekar and F. Sahin, "A survey on feature selection methods," *Computers & Electrical Engineering*, vol. 40, no. 1, pp. 16–28, 2014.

[7] W. W. Cohen, "Fast effective rule induction," in *Proceedings of the Twelfth International Conference on Machine Learning*, 1995, pp. 115–123.

[8] S. Dasgupta, M. L. Littman, and D. McAllester, "Pac generalization bounds for co-training," *Advances in neural information processing systems*, vol. 1, pp. 375–382, 2002.

[9] M. Dash, K. Choi, P. Scheuermann, and H. Liu, "Feature selection for clustering-a filter solution," in *Proceedings of the 2002 IEEE International Conference on Data Mining*. IEEE, 2002, pp. 115–122.

[10] M. Dash and H. Liu, "Feature selection for clustering," in *Pacific-Asia Conference on Knowledge Discovery and Data Mining*. Springer, 2000, pp. 110–121.

[11] A. P. Dawid, "Conditional independence in statistical theory," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 1–31, 1979.

[12] T. G. Dietterich, "Ensemble methods in machine learning," in *International Workshop on Multiple Classifier Systems*. Springer, 2000, pp. 1–15.

[13] J. G. Dy and C. E. Brodley, "Feature selection for unsupervised learning," *Journal of machine learning research*, vol. 5, no. Aug, pp. 845–889, 2004.

[14] V. Estivill-Castro, "Why so many clustering algorithms: a position paper," *ACM SIGKDD explorations newsletter*, vol. 4, no. 1, pp. 65–75, 2002.

[15] E. B. Fowlkes and C. L. Mallows, "A method for comparing two hierarchical clusterings," *Journal of the American statistical association*, vol. 78, no. 383, pp. 553–569, 1983.

[16] C. Fraley and A. E. Raftery, "Model-based clustering, discriminant analysis, and density estimation," *Journal of the American statistical Association*, vol. 97, no. 458, pp. 611–631, 2002.

[17] G. Fung, "A comprehensive overview of basic clustering algorithms," University of Wisconsin - Madison, 2001.

[18] D. Gamberger, M. Mihelčić, and N. Lavrač, "Multilayer clustering: a discovery experiment on country level trading data," in *International Conference on Discovery Science*. Springer, 2014, pp. 87–98.

[19] D. Gamberger, B. Ženko, A. Mitelpunkt, and N. Lavrač, "Multilayer clustering: biomarker driven segmentation of alzheimer's disease patient population," in *International Conference on Bioinformatics and Biomedical Engineering*. Springer, 2015, pp. 134–145.

[20] M. Gönen and E. Alpaydın, "Multiple kernel learning algorithms," *Journal of Machine Learning Research*, vol. 12, no. Jul, pp. 2211–2268, 2011.

[21] M. A. Hearst, S. T. Dumais, E. Osman, J. Platt, and B. Scholkopf, "Support vector machines," *IEEE Intelligent Systems and their Applications*, vol. 13, no. 4, pp. 18–28, 1998.

[22] N. Ilc, "Primerjava metod za razvrščanje vzorcev v gruče," diplomsko delo. Univerza v Ljubljani, Fakulteta za računalništvo in informatiko, 2009.

[23] R. Kohavi *et al.*, "A study of cross-validation and bootstrap for accuracy estimation and model selection," in *International Joint Conference on Artificial Intelligence*, vol. 14, no. 2, 1995, pp. 1137–1145.

[24] A. Kumar and H. Daumé, "A co-training approach for multi-view spectral clustering," in *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, 2011, pp. 393–400.

[25] J. Lee Rodgers and W. A. Nicewander, "Thirteen ways to look at the correlation coefficient," *The American Statistician*, vol. 42, no. 1, pp. 59–66, 1988.

[26] J. Li, K. Cheng, S. Wang, F. Morstatter, R. P. Trevino, J. Tang, and H. Liu, "Feature selection: A data perspective," *CoRR*, vol. abs/1601.07996, 2016.

[27] M. Lichman, "UCI machine learning repository," 2013. [Online]. Available: http://archive.ics.uci.edu/ml

[28] Y. Liu, Z. Li, H. Xiong, X. Gao, and J. Wu, "Understanding of internal clustering validation measures," in *2010 IEEE International Conference on Data Mining*. IEEE, 2010, pp. 911–916.

[29] M. A. Álvarez, L. Rosasco, and N. D. Lawrence, "Kernels for vector-valued functions: A review," *Foundations and Trends in Machine Learning*, vol. 4, no. 3, pp. 195–266, 2012.

[30] S. G. Mueller, M. W. Weiner, L. J. Thal, R. C. Petersen, C. R. Jack, W. Jagust, J. Q. Trojanowski, A. W. Toga, and L. Beckett, "Ways toward an early diagnosis in alzheimer's disease: the alzheimer's disease neuroimaging initiative (adni)," *Alzheimer's & Dementia*, vol. 1, no. 1, pp. 55–66, 2005.

[31] A. Y. Ng, M. I. Jordan, Y. Weiss *et al.*, "On spectral clustering: Analysis and an algorithm," *Advances in neural information processing systems*, vol. 2, pp. 849–856, 2002.

[32] W. S. Noble *et al.*, "Support vector machine applications in computational biology," *Kernel methods in computational biology*, pp. 71–92, 2004.

[33] L. Parida and N. Ramakrishnan, "Redescription mining: Structure theory and algorithms," in *Association for the Advancement of Artificial Intelligence*, vol. 5, 2005, pp. 837–844.

[34] D. Pfitzner, R. Leibbrandt, and D. Powers, "Characterization and evaluation of similarity measures for pairs of clusterings," *Knowledge and Information Systems*, vol. 19, no. 3, pp. 361–394, 2009.

[35] T. Polajnar, T. Damoulas, and M. Girolami, "Protein interaction sentence detection using multiple semantic kernels," *Journal of biomedical semantics*, vol. 2, no. 1, p. 1, 2011.

[36] N. Ramakrishnan, D. Kumar, B. Mishra, M. Potts, and R. F. Helm, "Turning cartwheels: an alternating algorithm for mining redescriptions," in *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2004, pp. 266–275.

[37] E. Rendón, I. Abundez, A. Arizmendi, and E. Quiroz, "Internal versus external cluster validation indexes," *International Journal of computers and communications*, vol. 5, no. 1, pp. 27–34, 2011.

[38] M. Robnik-Šikonja and I. Kononenko, "Theoretical and empirical analysis of relieff and rrelieff," *Machine learning*, vol. 53, no. 1-2, pp. 23–69, 2003.

[39] M. Robnik-Šikonja and I. Kononenko, "Explaining classifications for individual instances," *IEEE Transactions on Knowledge and Data Engineering*, vol. 20, no. 5, pp. 589–600, 2008.

[40] L. Rokach and O. Maimon, *Data mining with decision trees: theory and applications*. World scientific, 2014.

[41] F. Saeed, N. Salim, and A. Abdo, "Combining multiple clusterings of chemical structures using cluster-based similarity partitioning algorithm," *International journal of computational biology and drug design*, vol. 7, no. 1, pp. 31–44, 2014.

[42] K. R. Sakharkar, M. K. Sakharkar, and R. Chandra, *Post-Genomic Approaches in Drug and Vaccine Development*. River Publishers, 2015, vol. 5.

[43] V. Sindhwani, P. Niyogi, and M. Belkin, "A co-regularization approach to semi-supervised learning with multiple views," in *Proceedings of*

ICML workshop on learning with multiple views. Citeseer, 2005, pp. 74–79.

[44] A. Strehl and J. Ghosh, "Cluster ensembles—a knowledge reuse framework for combining multiple partitions," *Journal of Machine Learning Research*, vol. 3, no. Dec, pp. 583–617, 2002.

[45] S. Wagner and D. Wagner, "Comparing clusterings: an overview," karlsruhe Institute of Technology, Department of Informatics, 2007.

[46] D. Weenink, "Canonical correlation analysis," in *Proceedings of the Institute of Phonetic Sciences of the University of Amsterdam*, vol. 25. Citeseer, 2003, pp. 81–99.

[47] M. White, X. Zhang, D. Schuurmans, and Y.-l. Yu, "Convex multi-view subspace learning," in *Advances in Neural Information Processing Systems*, 2012, pp. 1673–1681.

[48] C. Xu, D. Tao, and C. Xu, "A survey on multi-view learning," *CoRR*, vol. abs/1304.5634, 2013.

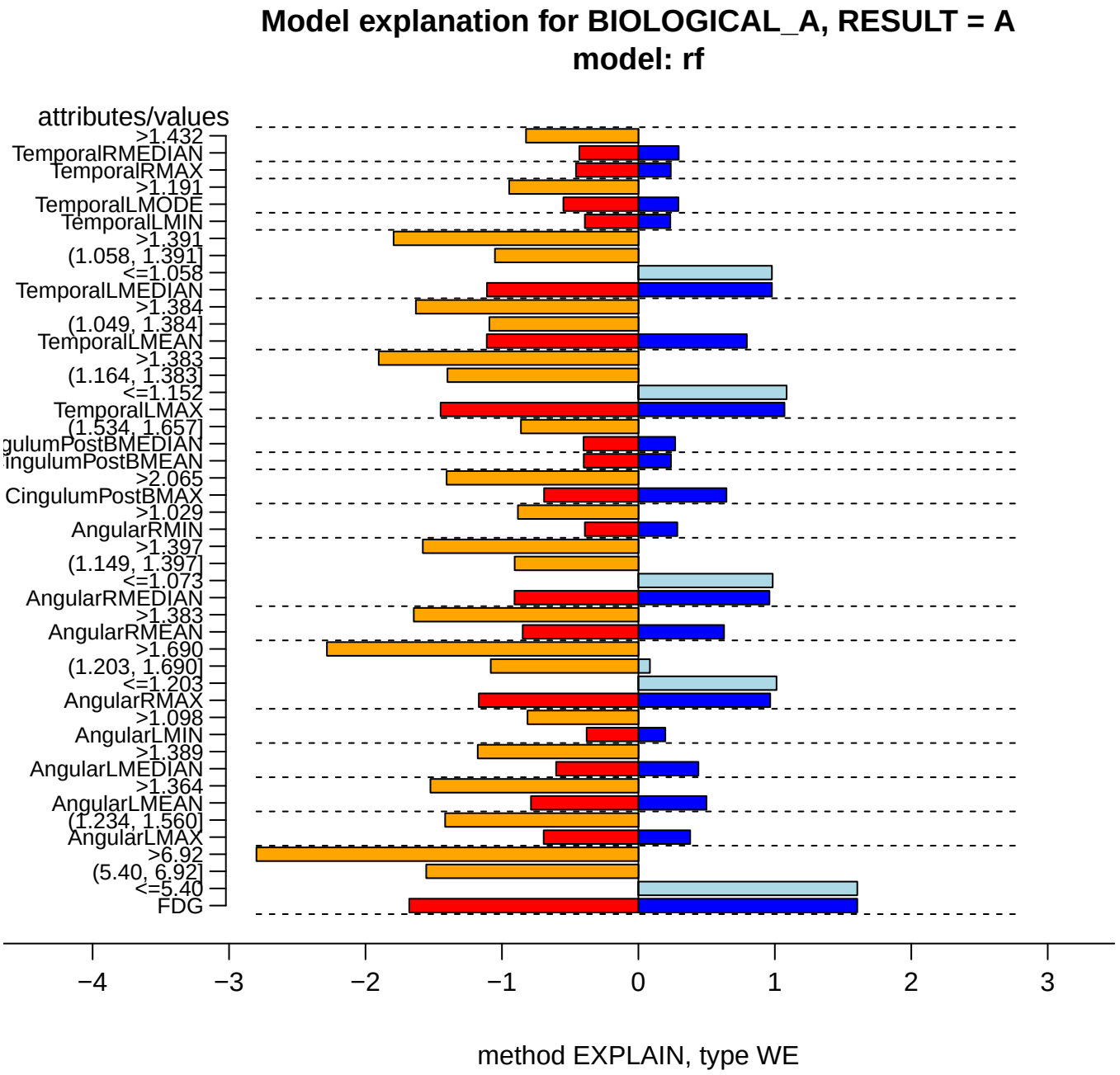
[49] T. Zhang, R. Ramakrishnan, and M. Livny, "Birch: an efficient data clustering method for very large databases," in *ACM Sigmod Record*, vol. 25, no. 2. ACM, 1996, pp. 103–114.

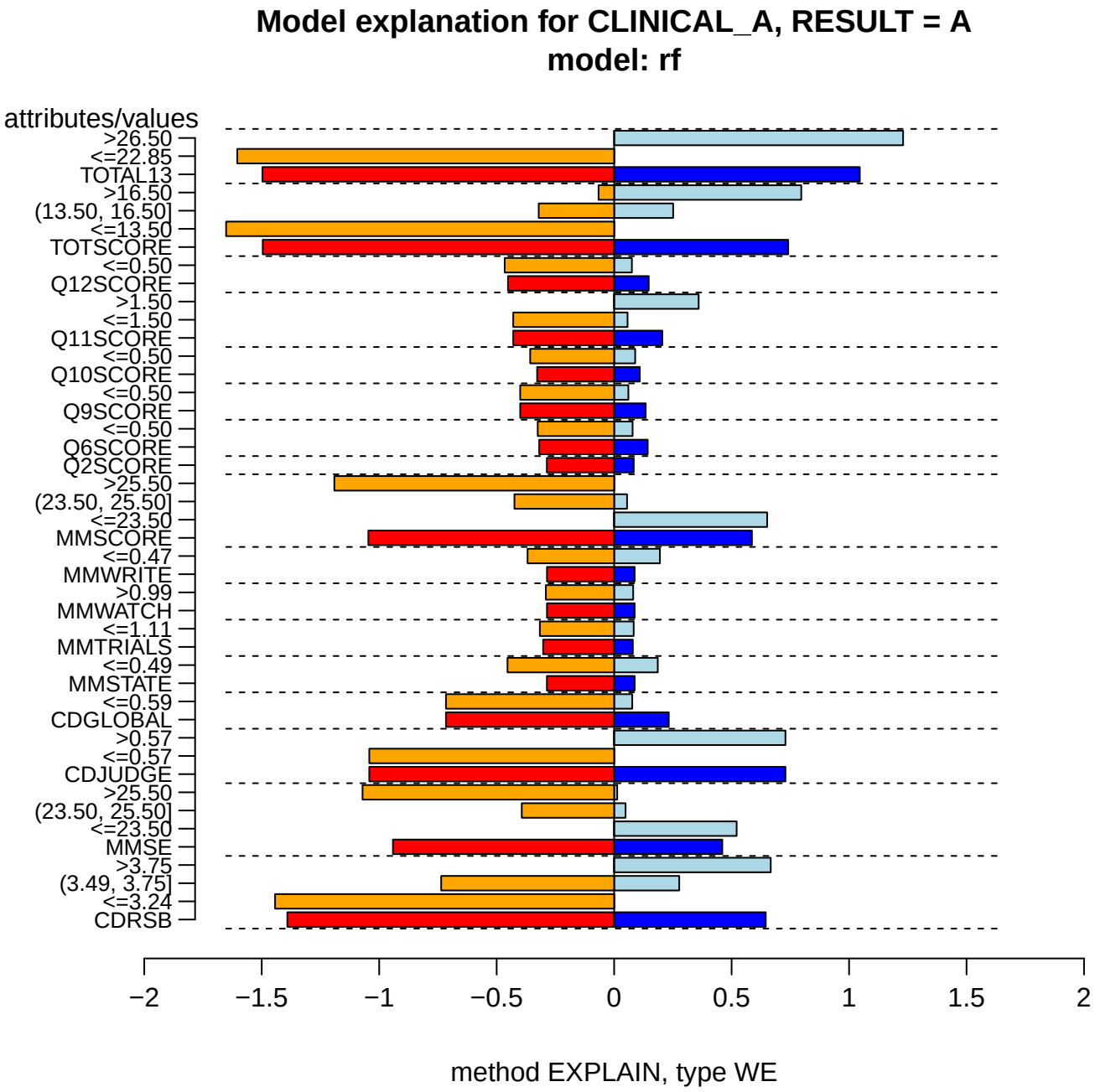
[50] X. Zhu, "Semi-supervised learning," in *Encyclopedia of machine learning*. Springer, 2011, pp. 892–897.

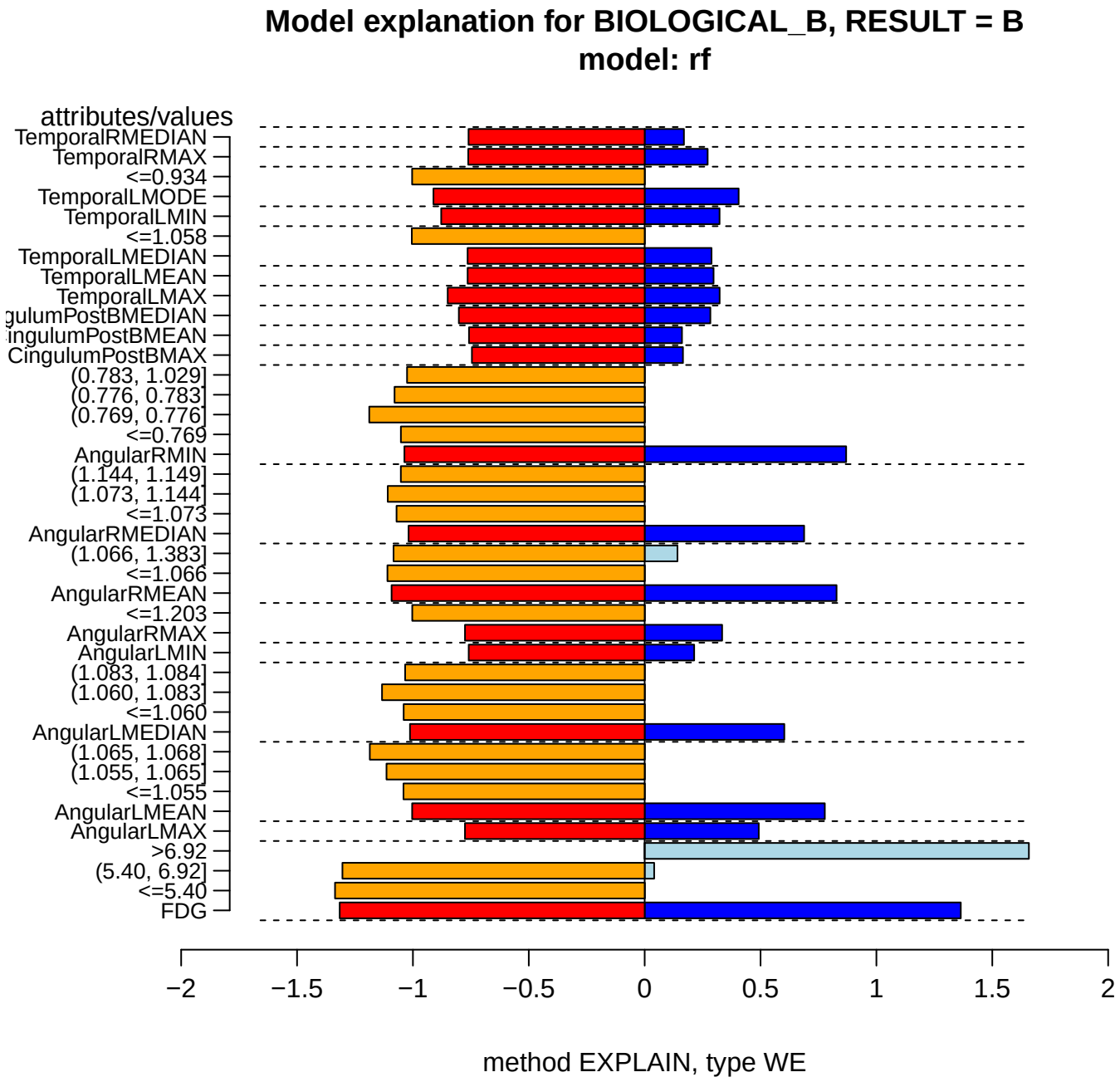
Dodatek A

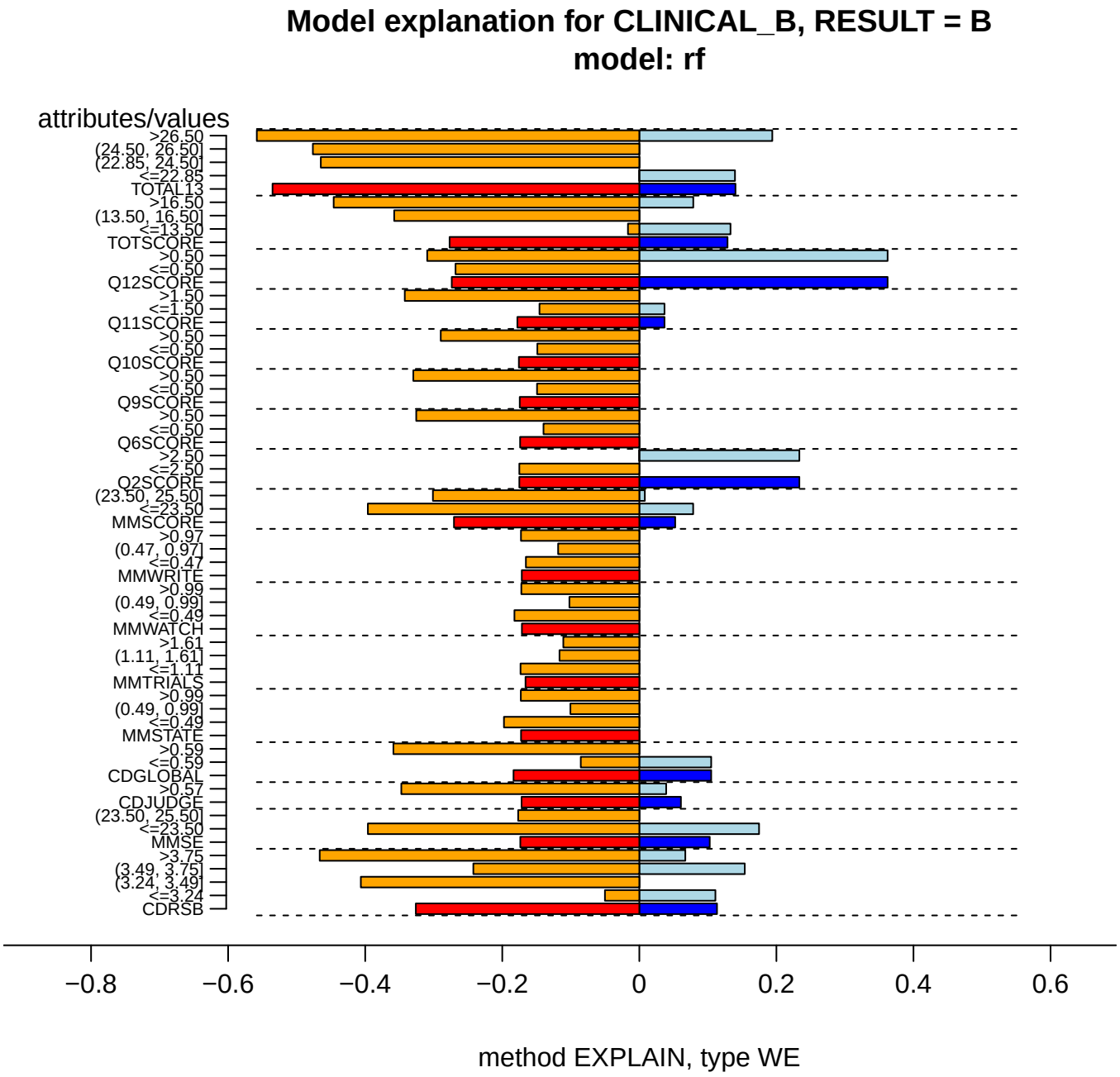
Splošne razlage gruč pacientov

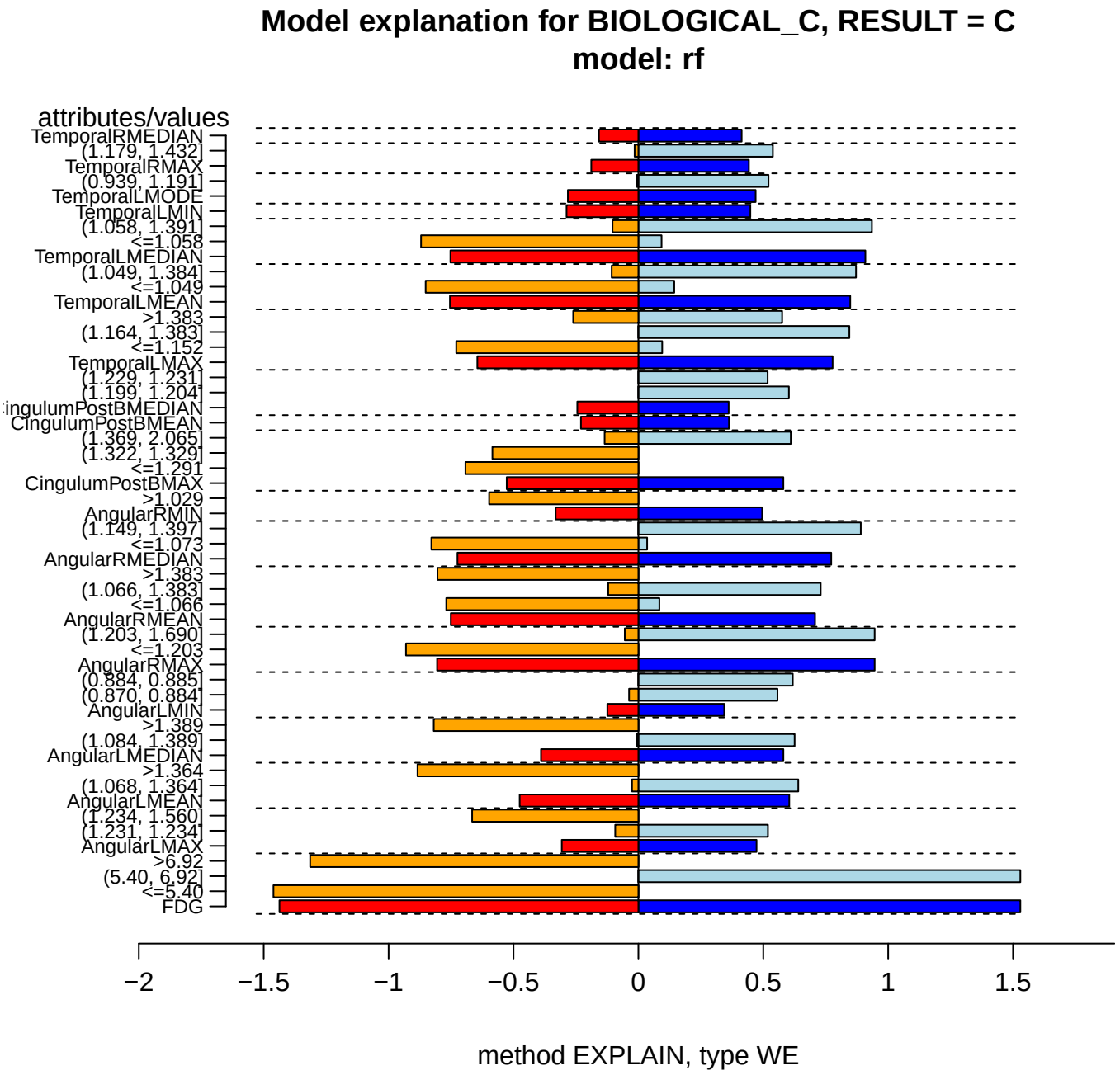
Spodnje slike vizualizirajo vpliv značilk na uvrstitev primerov v gruče, ki smo jih dobili z izvajanjem predlagane metodologije na ADNI podatkih. Vsaka gruča je razložena posebej z biološkimi značilkami (slike, ki se začnejo z "Model explanation for BIOLOGICAL") in kliničnimi značilkami (slike, ki se začnejo z "Model explanation for CLINICAL")

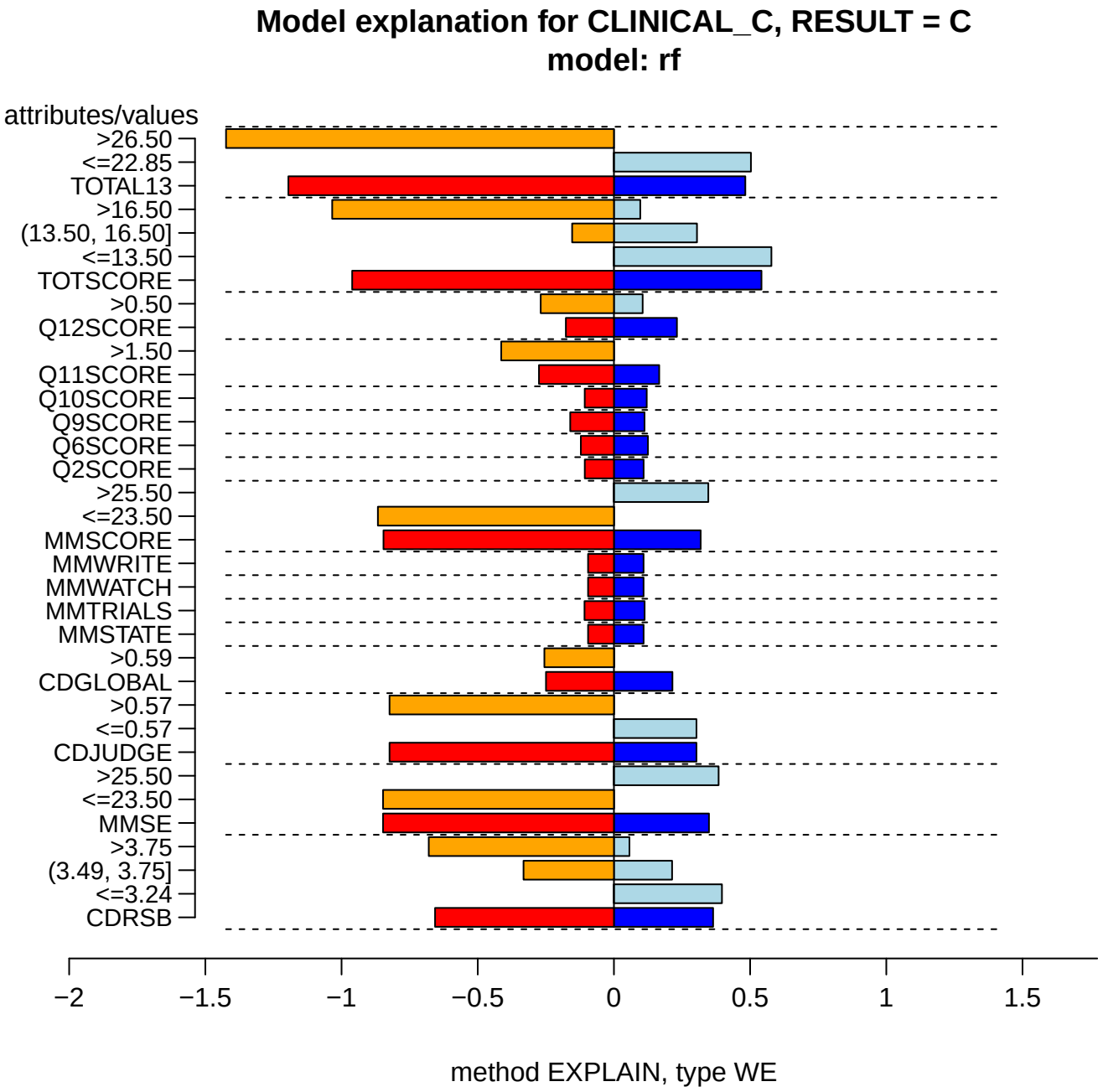

Model explanation for BIOLOGICAL_A, RESULT = A
model: rf
attributes/values
>1.432
TemporalRMEDIAN
TemporalRMAX
>1.191
TemporalLMODE
TemporalLMIN
>1.391
(1.058, 1.391]
<=1.058
TemporalLMEDIAN
>1.384
(1.049, 1.384]
TemporalLMEAN
>1.383
(1.164, 1.383]
<=1.152
TemporalLMAX
(1.534, 1.657]
gulumPostBMEDIAN
ingulumPostBMEAN
>2.065
CingulumPostBMAX
>1.029
AngularRMIN
>1.397
(1.149, 1.397]
<=1.073
AngularRMEDIAN
>1.383
AngularRMEAN
>1.690
(1.203, 1.690]
<=1.203
AngularRMAX
>1.098
AngularLMIN
>1.389
AngularLMEDIAN
>1.364
AngularLMEAN
(1.234, 1.560]
AngularLMAX
>6.92
(5.40, 6.92]
<=5.40
FDG
−4
−3
−2
−1
0
1
2
3
method EXPLAIN, type WE


Model explanation for CLINICAL_A, RESULT = A
model: rf
attributes/values
>26.50
<=22.85
TOTAL13
>16.50
(13.50, 16.50]
<=13.50
TOTSCORE
<=0.50
Q12SCORE
>1.50
<=1.50
Q11SCORE
<=0.50
Q10SCORE
<=0.50
Q9SCORE
<=0.50
Q6SCORE
Q2SCORE
>25.50
(23.50, 25.50]
<=23.50
MMSCORE
<=0.47
MMWRITE
>0.99
MMWATCH
<=1.11
MMTRIALS
<=0.49
MMSTATE
<=0.59
CDGLOBAL
>0.57
<=0.57
CDJUDGE
>25.50
(23.50, 25.50]
<=23.50
MMSE
>3.75
(3.49, 3.75]
<=3.24
CDRSB
−2
−1.5
−1
−0.5
0
0.5
1
1.5
2
method EXPLAIN, type WE


Model explanation for BIOLOGICAL_B, RESULT = B
model: rf
attributes/values
TemporalRMEDIAN
TemporalRMAX
<=0.934
TemporalLMODE
TemporalLMIN
<=1.058
TemporalLMEDIAN
TemporalLMEAN
TemporalLMAX
gulumPostBMEDIAN
ingulumPostBMEAN
CingulumPostBMAX
(0.783, 1.029]
(0.776, 0.783]
(0.769, 0.776]
<=0.769
AngularRMIN
(1.144, 1.149]
(1.073, 1.144]
<=1.073
AngularRMEDIAN
(1.066, 1.383]
<=1.066
AngularRMEAN
<=1.203
AngularRMAX
AngularLMIN
(1.083, 1.084]
(1.060, 1.083]
<=1.060
AngularLMEDIAN
(1.065, 1.068]
(1.055, 1.065]
<=1.055
AngularLMEAN
AngularLMAX
>6.92
(5.40, 6.92]
<=5.40
FDG
−2
−1.5
−1
−0.5
0
0.5
1
1.5
2
method EXPLAIN, type WE


Model explanation for CLINICAL_B, RESULT = B
model: rf
attributes/values
>26.50
(24.50, 26.50]
(22.85, 24.50]
<=22.85
TOTAL13
>16.50
(13.50, 16.50]
<=13.50
TOTSCORE
>0.50
<=0.50
Q12SCORE
>1.50
<=1.50
Q11SCORE
>0.50
<=0.50
Q10SCORE
>0.50
<=0.50
Q9SCORE
>0.50
<=0.50
Q6SCORE
>2.50
<=2.50
Q2SCORE
(23.50, 25.50]
<=23.50
MMSCORE
>0.97
(0.47, 0.97]
<=0.47
MMWRITE
>0.99
(0.49, 0.99]
<=0.49
MMWATCH
>1.61
(1.11, 1.61]
<=1.11
MMTRIALS
>0.99
(0.49, 0.99]
<=0.49
MMSTATE
>0.59
<=0.59
CDGLOBAL
>0.57
CDJUDGE
(23.50, 25.50]
<=23.50
MMSE
>3.75
(3.49, 3.75]
(3.24, 3.49]
<=3.24
CDRSB
−0.8
−0.6
−0.4
−0.2
0
0.2
0.4
0.6
method EXPLAIN, type WE


Model explanation for BIOLOGICAL_C, RESULT = C
model: rf
attributes/values
TemporalRMEDIAN
(1.179, 1.432]
TemporalRMAX
(0.939, 1.191]
TemporalLMODE
TemporalLMIN
(1.058, 1.391]
<=1.058
TemporalLMEDIAN
(1.049, 1.384]
<=1.049
TemporalLMEAN
>1.383
(1.164, 1.383]
<=1.152
TemporalLMAX
(1.229, 1.231]
(1.199, 1.204]
ingulumPostBMEDIAN
CingulumPostBMEAN
(1.369, 2.065]
(1.322, 1.329]
<=1.291
CingulumPostBMAX
>1.029
AngularRMIN
(1.149, 1.397]
<=1.073
AngularRMEDIAN
>1.383
(1.066, 1.383]
<=1.066
AngularRMEAN
(1.203, 1.690]
<=1.203
AngularRMAX
(0.884, 0.885]
(0.870, 0.884]
AngularLMIN
>1.389
(1.084, 1.389]
AngularLMEDIAN
>1.364
(1.068, 1.364]
AngularLMEAN
(1.234, 1.560]
(1.231, 1.234]
AngularLMAX
>6.92
(5.40, 6.92]
<=5.40
FDG
−2
−1.5
−1
−0.5
0
0.5
1
1.5
method EXPLAIN, type WE

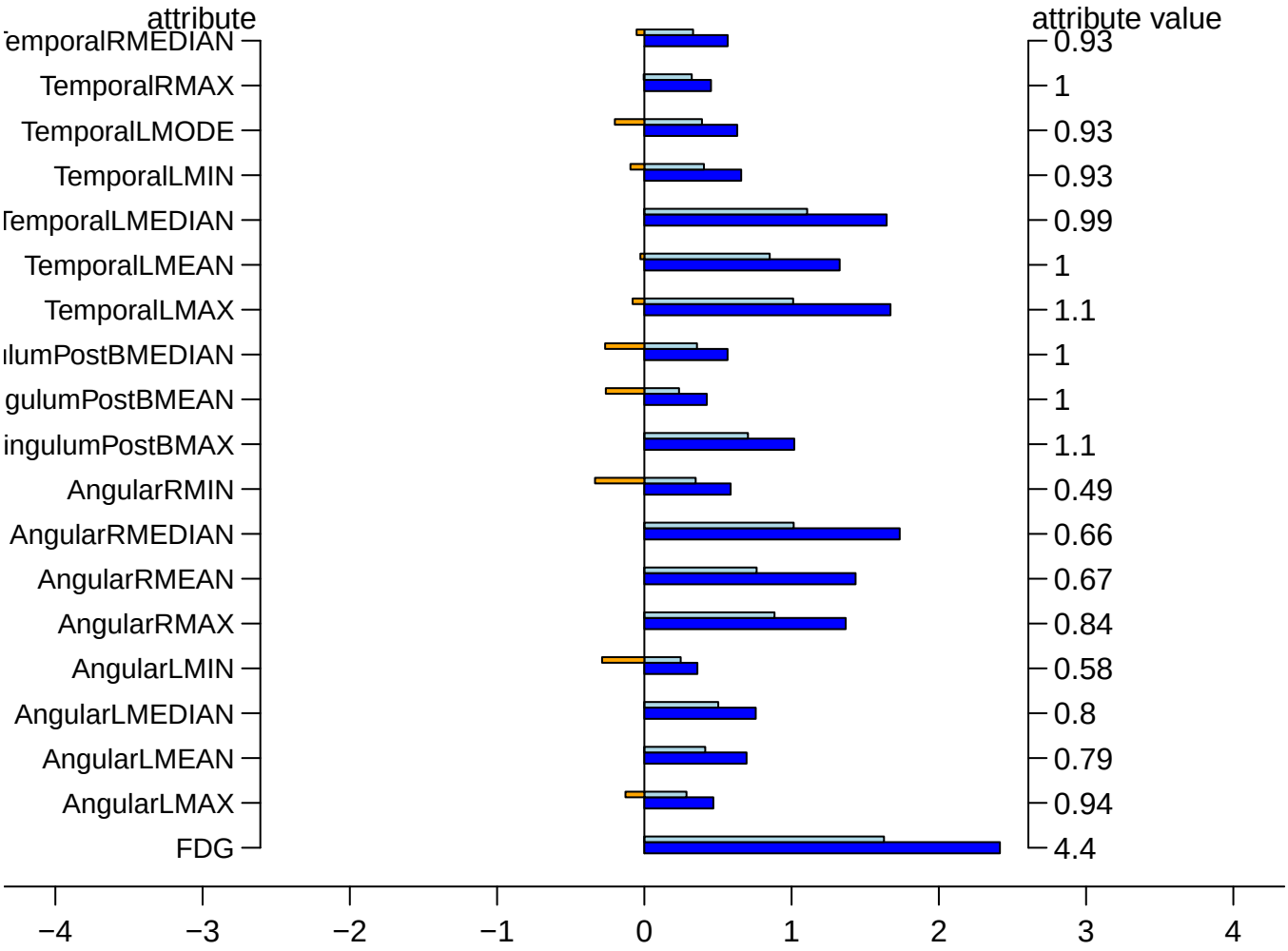

Model explanation for CLINICAL_C, RESULT = C
model: rf
attributes/values
>26.50
<=22.85
TOTAL13
>16.50
(13.50, 16.50]
<=13.50
TOTSCORE
>0.50
Q12SCORE
>1.50
Q11SCORE
Q10SCORE
Q9SCORE
Q6SCORE
Q2SCORE
>25.50
<=23.50
MMSCORE
MMWRITE
MMWATCH
MMTRIALS
MMSTATE
>0.59
CDGLOBAL
>0.57
<=0.57
CDJUDGE
>25.50
<=23.50
MMSE
>3.75
(3.49, 3.75]
<=3.24
CDRSB
−2
−1.5
−1
−0.5
0
0.5
1
1.5
method EXPLAIN, type WE

Dodatek B

Razlage gruč pacientov za posamezne primere

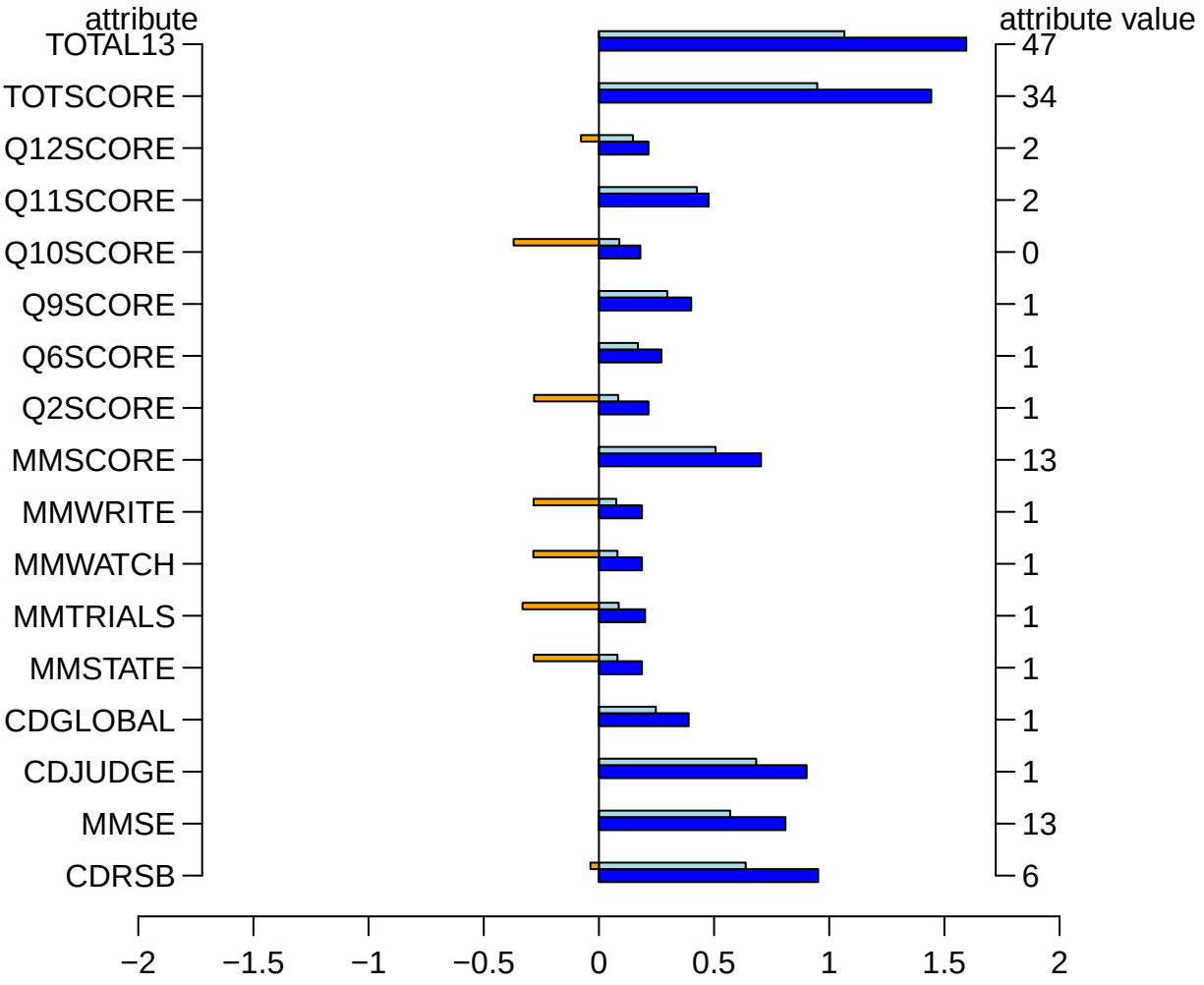
Spodnje slike za posamezne zanimive primere vizualizirajo vpliv značilk na uvrstitev primerov v gruče, ki smo jih dobili z izvajanjem predlagane metodologije na ADNI podatkih. Vsaka gruča je razložena posebej z biološkimi značilkami (slike, ki se začnejo z "Explaining prediction for BIO") in kliničnimi značilkami (slike, ki se začnejo z "Explaining prediction for CLINICAL")

Explaining prediction for BIO INSTANCE FOR CLASS A, RESULT = A
instance: 19, model: rf



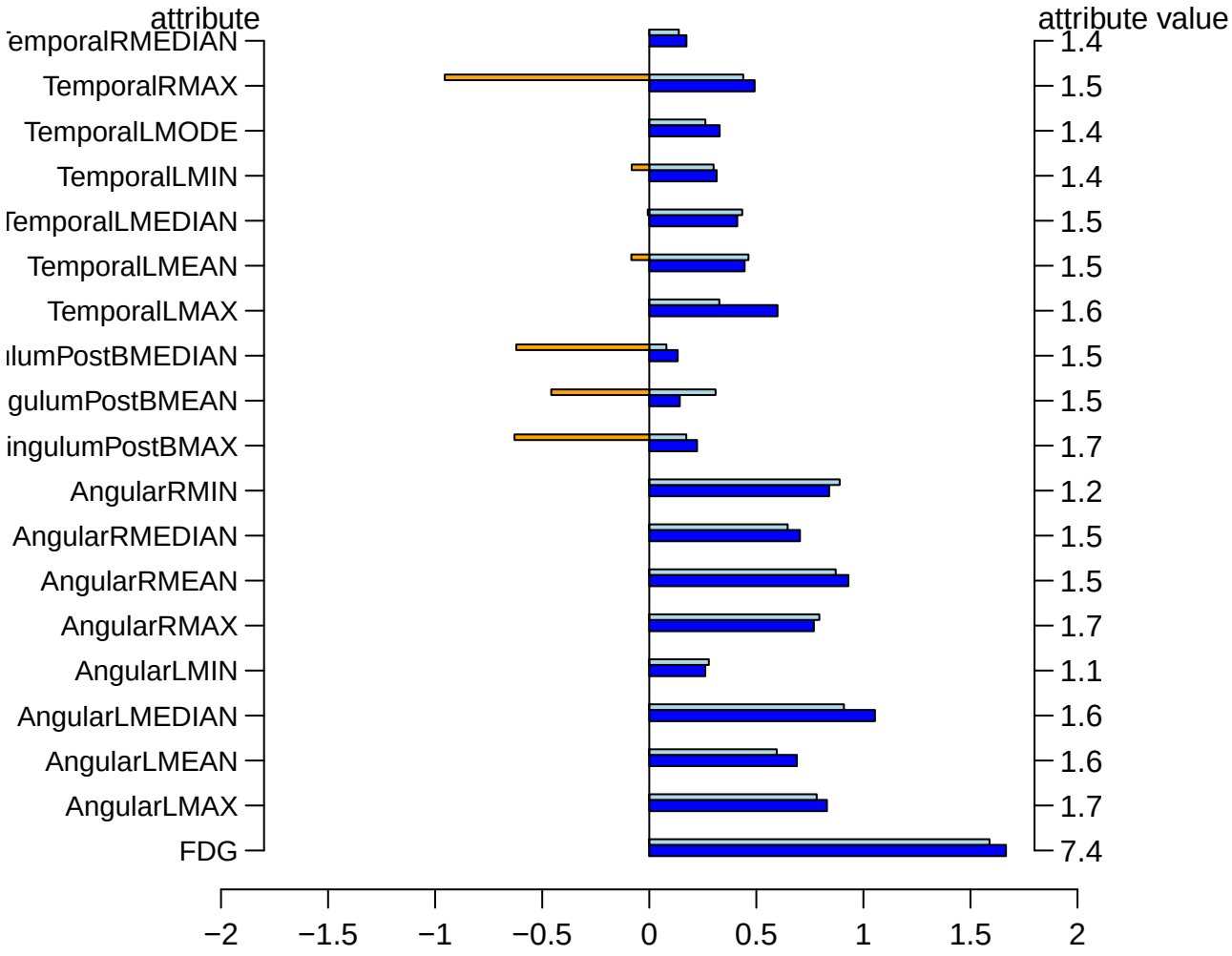
method EXPLAIN, type WE
p(RESULT=A) = 0.95; true RESULT=A

Explaining prediction for CLINICAL INSTANCE FOR CLASS A, RESULT = A
instance: 4, model: rf



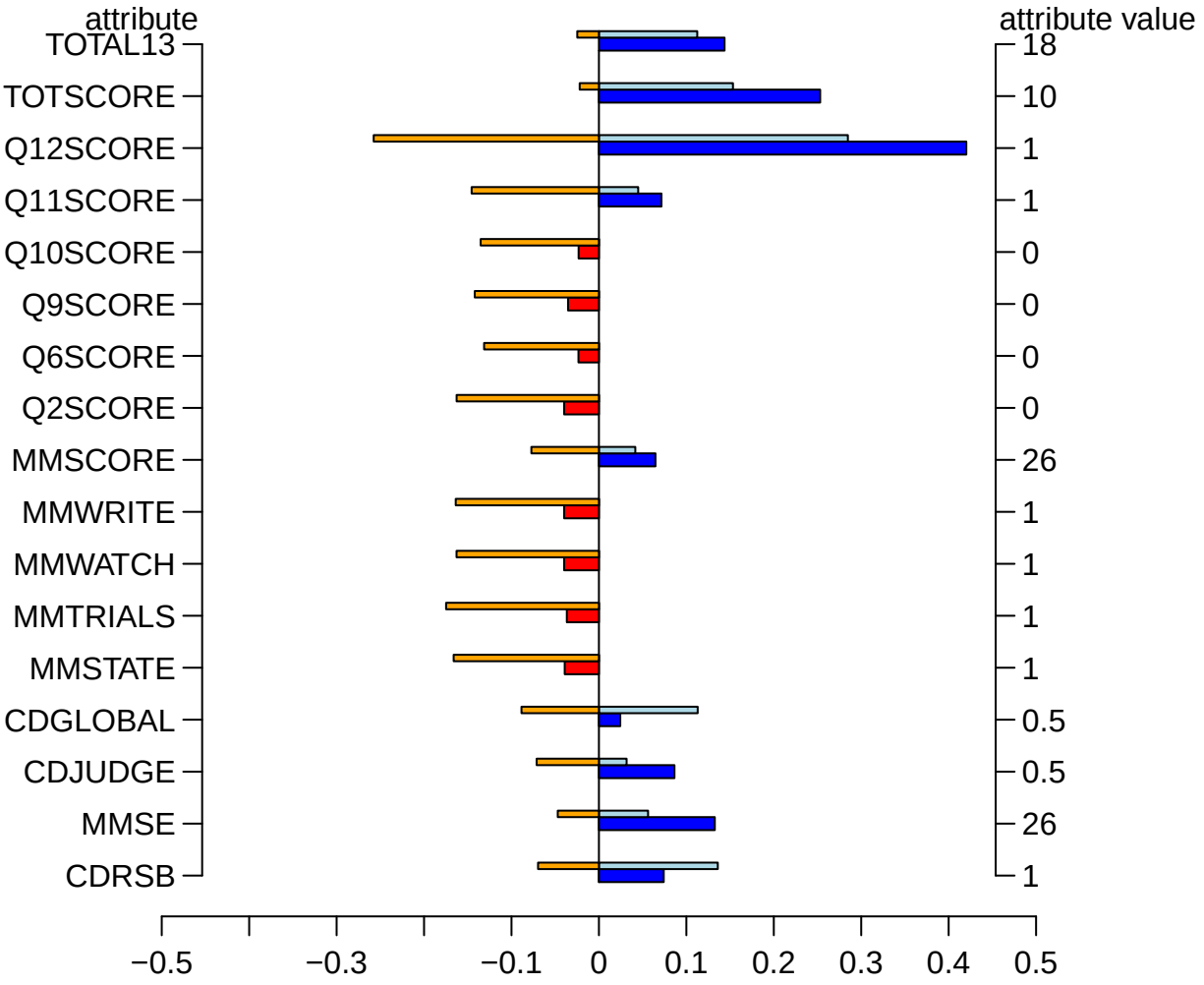
method EXPLAIN, type WE
p(RESULT=A) = 0.90; true RESULT=A

Explaining prediction for BIO INSTANCE FOR CLASS B, RESULT = B
instance: 15, model: rf



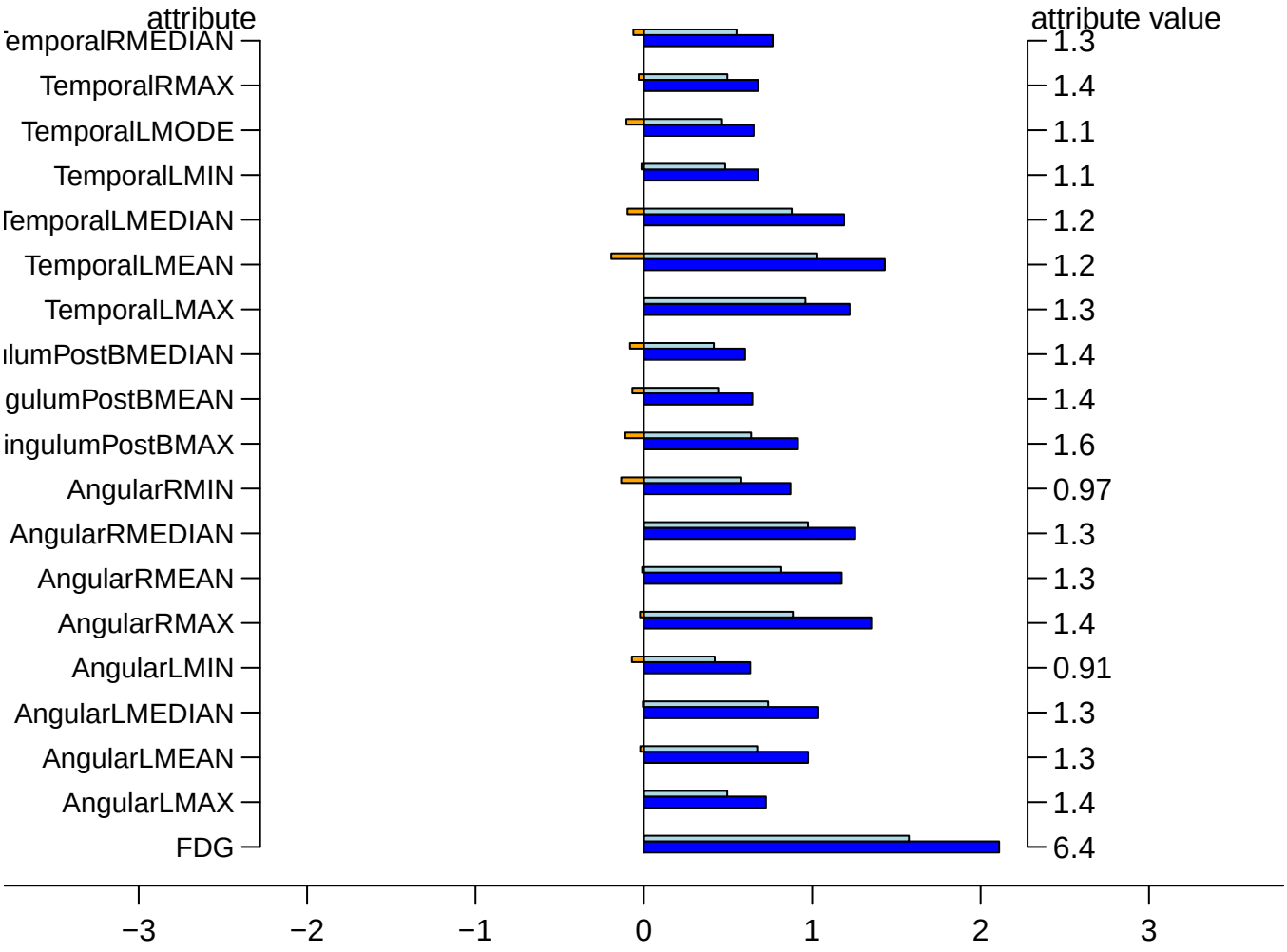
method EXPLAIN, type WE
p(RESULT=B) = 0.84; true RESULT=B

Explaining prediction for CLINICAL INSTANCE FOR CLASS B, RESULT = B
instance: 33, model: rf



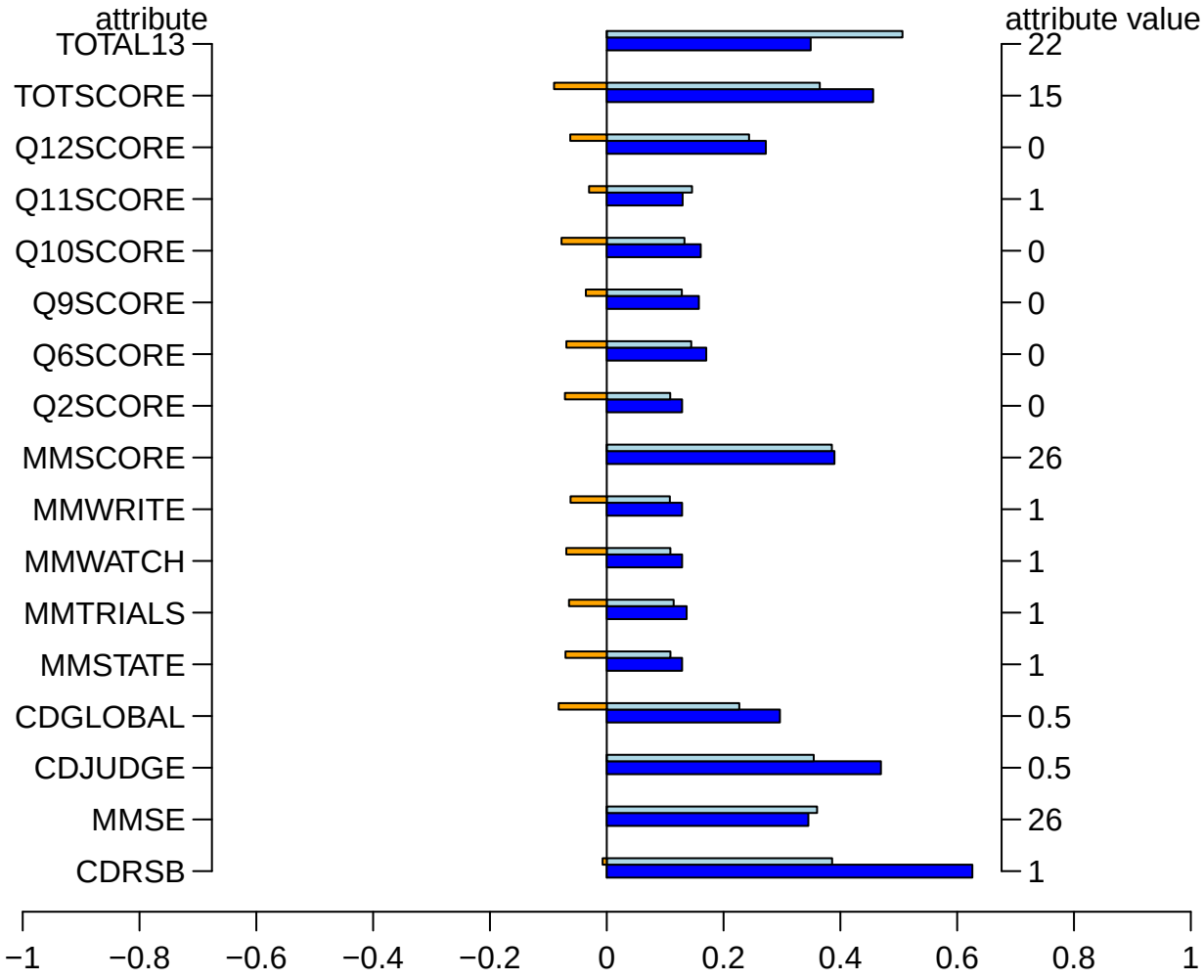
method EXPLAIN, type WE
p(RESULT=B) = 0.16; true RESULT=B

Explaining prediction for BIO INSTANCE FOR CLASS C, RESULT = C
instance: 41, model: rf



method EXPLAIN, type WE
p(RESULT=C) = 0.97; true RESULT=C

Explaining prediction for CLINICAL INSTANCE FOR CLASS C, RESULT = C
instance: 13, model: rf



method EXPLAIN, type WE
p(RESULT=C) = 0.86; true RESULT=C

Dodatek C

Opis značilk

Priloga vsebuje opis značilk pridobljenih z izbiro značilk na združeni ADNI podatkovni zbirki.

C.1 Klinične značilke

CDRSB Clinical Dementia Rating-Sum of the Boxes scale je klinični test, ki preverja kategorije: spomin, orientacija, odnosi, presoja, reševanje problemov ter sposobnost opravljanja hobijev.

Vsaka kategorija se oceni od 0-3, kjer je 0 najboljša in 3 najbolj kritična ocena. Če ocene vseh kategorij seštejemo, dobimo CDR sum of boxes (CDRSB), ki ima vrednost največ 18 (6*3), kar je najbolj kritičen možen rezultat.

CDJUDGE je del kliničnega testa Clinical Dementia Rating, ki preverja presojo in reševanje problemov. 3 je najslabša ocena in 0 najboljša.

CDGLOBAL je ocena kliničnega testa Clinical Dementia Rating, ki rezultat izračuna na podlagi uteženih rezultatov kategorij testa CDR.

MMSE je ocena kliničnega testa Mini–Mental State Examination (MMSE). Točke gredo od 0 do 30. Večja ocena nakazuje boljše kognitivne sposobnosti.

MMSTATE je značilka testa Mini–Mental State Examination (MMSE), ki označuje pravilnost/nepravilnost odgovora na vprašanje »v kateri provinci se nahajamo«. Vrednost 1 je za pravilen odgovor, 0 za nepravilen.

MMWATCH je značilka testa Mini–Mental State Examination (MMSE), ki označuje pravilnost/nepravilnost odgovora na vprašanje »kaj je to?«, ko s prstom pokažemo na ročno uro.

MMTRIALS je značilka testa Mini–Mental State Examination (MMSE), ki , kolikokrat je pacient ponovil 13. Značilka je nerelevantna za namen raziskave. Tudi pri vizualizaciji vpliva značilk ne igra pomembne vloge.

MMWRITE je značilka testa Mini–Mental State Examination (MMSE). Od pacienta se zahteva, da na list papirja napiše poljuben stavek. Značilka ima vrednost 1, če mu uspe, 0, če ne.

MMSCORE je ocena kliničnega testa Mini–Mental State Examination (MMSE). Točke gredo od 0 do 30. Večja ocena nakazuje boljše kognitivne sposobnosti.

Q2SCORE je ocena 2. vprašanja kliničnega testa Alzheimer's Disease Assessment Scale-Cognitive (ADAS-Cog) – Commands, ki ocenjuje sposobnost sledenja ukazom. Na primer, od pacienta se zahteva, da s prstom pokaže na strop. Vrednosti gredo od 0 do 5 (več je slabše).

Q6SCORE je ocena 6. vprašanja kliničnega testa Alzheimer's Disease Assessment Scale-Cognitive (ADAS-Cog) – Ideational Praxis, ki ocenjuje sposobnosti opravljanja znanih, a kompleksnih opravil, kot je pošiljanje pisma. Pacient mora papir vstaviti v kuverto, jo zalepiti in jo nasloviti na svoj naslov. Na koncu mora nalepiti znamko. Vrednosti gredo od 0 do 5 (več je slabše).

Q9SCORE je ocena 9. vprašanja kliničnega testa Alzheimer's Disease Assessment Scale-Cognitive (ADAS-Cog) – Remembering Test Instructions, ki ocenjuje sposobnost ponovitve navodil, ki so bile pacientu razložena ne dolgo nazaj. Vrednosti gredo od 0 do 5 (več je slabše).

Q10SCORE je ocena 10. vprašanja kliničnega testa Alzheimer's Disease Assessment Scale-Cognitive (ADAS-Cog) – Spoken Language, ki ocenjuje sposobnost govora pacienta. Vrednosti gredo od 0 do 5 (več je slabše).

Q11SCORE je ocena 11. vprašanja kliničnega testa Alzheimer's Disease Assessment Scale-Cognitive (ADAS-Cog) – Word finding, ki ocenjuje

sposobnost uporabe ustreznih besed pri spontanem govoru. Vrednosti gredo od 0 do 5 (več je slabše).

Q12SCORE je ocena 12. vprašanja kliničnega testa Alzheimer's Disease Assessment Scale-Cognitive (ADAS-Cog) – Comprehension, ki ocenjuje sposobnost razumevanja govora. Vrednosti gredo od 0 do 5 (več je slabše).

TOTSCORE je seštevek ocen posameznih preizkusov kliničnega testa Alzheimer's Disease Assessment ScaleCognitive (ADAS-Cog). Vrednosti gredo od 0 do 70. Večje število, bolj kritičen je pacient.

TOTAL13 je seštevek ocen posameznih preizkusov kliničnega variacije testa ADAS - Alzheimer's Disease Assessment Scale-Cognitive (modified ADAS-Cog). Ocene gredo od 0 do 85. Večje število, bolj kritičen je pacient.

C.2 Biološke značilke

Metoda izbire značilk je med biološkimi značilkami izbirala le značilke FDG-PET raziskave Berkeley FDG. Predstavljajo mediano (MEDIAN), modus (MODE), največjo (MAX), najmanjšo (MIN) in srednjo (MEAN) vrednost fludeoksiglukoze z uporabo pozitronske emisijske tomografije za naslednjih pet področij:

- Left Angular Gyrus
- Right Angular Gyrus
- Bilateral Posterior Cingular
- Left Inferior Temporal Gyrus
- Right Inferior Temporal Gyrus

Izbrane biološke značilke so: **FDG, AngularLeftMAX, AngularLeftMEAN, AngularLeftMEDIAN, AngularLeftMIN, AngularRightMAX, AngularRightMEAN, AngularRightMEDIAN, AngularRightMIN,**

CingulumPostBilateralMAX, CingulumPostBilateralMEAN, CingulumPostBilateralMEDIAN, TemporalLeftMAX, TemporalLeftMEAN, TemporalLeftMEDIAN, TemporalLeftMIN, TemporalLeftMODE, TemporalRightMAX, TemporalRightMEDIAN

Vse značilke v imenu vsebujejo stran (L - left, R - right) in tip vrednosti (MEAN, MODE, ...). Izjema je značilka FDG, ki predstavlja povprečno vrednost vseh pet področij.

Primer razlage imena značilke: **AngularLMEAN** predstavlja srednjo (MEAN) vrednost s področja Left Angular Gyrus.

Dodatek D

Doprinos k odprtokodnim knjižnicam za izbiro značilk

Pri implementaciji predlagane metodologije smo za izvajanje obstoječih algoritmov izbire značilk uporabljali knjižnico scikit-feature [26]. Pri testiranju izvajanja implementacije algoritma ReliefF smo opazili, da implementacija za izbiro m naključnih primerov uporablja vzorčenje s ponavljanjem. Tako tudi pri učenju na vseh učnih primerih (m = število primerov) nismo vedno dobili istih rezultatov. Implementacija algoritma z vzorčenjem s ponavljanjem je napačna, saj dopušča možnost, da so isti primeri večkrat izbrani za učenje. Omenjeno napako smo popravili in popravek poslali na GitHub repozitorij knjižnice, kjer je bil popravek potrjen in je prisoten v zadnji javno dostopni različici knjižnice.